

**COLAB : PROPOSITION D'UNE PLATEFORME ACADÉMIQUE
COOPÉRATIVE, COLLABORATIVE, INTERDISCIPLINAIRE ET
RÉFLEXIVE D'ANALYSE COMPORTEMENTALE EN
ENVIRONNEMENT INTELLIGENT**

par

Jules Randolph

Mémoire présenté au Département d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES
UNIVERSITÉ DE SHERBROOKE

Sherbrooke, Québec, Canada, 28 septembre 2017

Le 28 septembre 2017

le jury a accepté le mémoire de Monsieur Jules Randolph dans sa version finale.

Membres du jury

Professeur Hélène Pigot
Directrice de recherche
Département d'informatique

Professeur Sylvain Giroux
Membre interne
Département d'informatique

Professeur Richard St-Denis
Président-rapporteur
Département d'informatique



Ce mémoire est mis à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale 2.5 Canada (CC BY-NC 2.5)¹. Les diagrammes créés pour le présent rapport ont été réalisés avec PlantUML et LibreOffice Draw. Les maquettes graphiques ont été dessinées à l'aide de Inkscape.

1. Pour voir une copie de cette licence, visitez <http://creativecommons.org/licenses/by-nc-nd/2.5/ca/> ou écrivez à Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Sommaire

Les perspectives de collaboration sont aujourd’hui démultipliées par l’irruption des nouvelles technologies dans nos sociétés. Mais dans le domaine de la recherche, des solutions tardent encore à émerger. Par exemple, les outils spécialisés d’assistance à l’**Analyse Comportementale (AC)** par codage vidéo ne tirent pas parti d’un déploiement en ligne. COLAB, qui prend la forme d’une application web, offre à divers chercheurs la possibilité de collaborer autour d’un ensemble de données d’expérimentation récoltées en appartement « intelligent ». Dans un premier temps, l’approche privilégiée est l’**AC**, mais la plateforme se veut évolutive et capable d’accueillir, à terme, des modules supportant un vaste champ d’expertises afin de constituer un cadre fécond pour l’interdisciplinarité. Un modèle de gouvernance inspiré du coopérativisme des plateformes est adjoint à COLAB pour proposer une exploitation semi-commerciale dont les bénéfices sont mécaniquement réinjectés dans la recherche.

L’identification du périmètre du problème et sa décomposition sont l’objet du **chapitre 3**, dans lequel nous proposons d’étudier les processus associés à l’**AC** et les expérimentations en environnement intelligent. Le **chapitre 4** présente les prototypes réalisés dans une démarche exploratoire ainsi que certains résultats de mesures ergonomiques. Enfin, des spécifications partielles sont offertes dans le **chapitre 5** pour projeter la solution dans une forme plus complète.

Mots-clés: collaboration, coopérativisme des plateformes, analyse comportementale, logiciel, interdisciplinarité, application web

Remerciements

Je veux remercier en premier lieu ma directrice de recherche, la professeure H  l  ne Pigot, pour ses pr  cieux conseils et la qualit   de son soutien. Je tiens    remercier avec toute mon affection mes parents, qui m’ont toujours t  moign   d’un soutien inconditionnel dans mes choix d’  tude et m’ont accompagn   tout au long de mon cheminement. Et aussi, tous ces chers coll  gues et amis, parmi lesquels Mathieu Gagnon, pour son accueil au Qu  bec et ses suggestions pertinentes sur COLAB, Camila Oliveira pour son soutien, son aide pour la soutenance et son ind  fectible bonne humeur, Christina Khnaisser pour ses excellentes suggestions de lecture et ses conseils de qualit   en ing  nierie des exigences, Damien Lockner pour ses remarques avis  es en ergonomie et en *design*, Wathek Bellah Loued pour m’avoir aiguill   vers des technologies web innovantes, Hubert Kenfack Ngankam pour m’avoir partag   sa vision d’un mod  le de t  ches et Yannick Drolet pour ses nombreux retours et sans qui je n’aurai pas pu r  parer mon Dell XPS 13 avec lequel j’ai r  dig   ce m  moire.

Je remercie aussi tous les collaborateurs du projet DEI    Montr  al, parmi lesquels Nathalie Bier sans qui COLAB n’aurait pas pu   tre prototyp  , Gabrielle Beaudoin Dion pour m’avoir expliqu   la d  marche d’  laboration de grilles d’analyse puis Maxime Lussier et Guillaume Paquette pour leurs retours d’utilisation sur COLAB.

Enfin, je dois remercier Pierre-Yves Groussard et Marc Chevalaz pour leur aide de qualit   avec l’architecture syst  me du **DOMotique et informatique Mobile    l’Universit   de Sherbrooke (DOMUS)**, puis Joan Montoya pour son audit de s  curit   particuli  rement efficace. Et tous ceux que je n’ai pas mentionn  s ici mais qui ont offert aide, soutien et bonne humeur, notamment    l’occasion de parties de babyfoot endiabl  es !

Table des matières

Licence	ii
Sommaire	iii
Remerciements	v
Table des matières	vii
Liste d’acronymes et d’abréviations	xiii
Liste des tableaux	xv
Table des figures	xvii
Glossaire technique	xxiii
Dictionnaire de domaine univoque	xxix
Introduction	1
1 Contexte global.....	1
2 Contexte local	2
1 État de l’art	5
1.1 Pluri, inter et transdisciplinarité	6
1.2 Solutions d’assistance à l’analyse comportementale par codage vidéo.....	8
1.2.1 Description générale.....	8
1.2.2 Analyse fonctionnelle détaillée	9
1.2.3 Comparaison	11
1.3 Le profil des AVQ, une méthodologie clinique d’évaluation de l’autonomie ...	13

1.3.1	Introduction	13
1.3.2	Mise en situation	14
1.3.3	Entrevue.....	16
1.3.4	Applicabilité au dépistage de la maladie d'Alzheimer	16
1.4	Systèmes de contrôle d'accès	18
1.4.1	Typologie des systèmes d'accès	18
1.4.2	Role Based Access Control	20
1.5	Coopérativisme des plateformes.....	25
1.5.1	Caractéristiques des plateformes coopératives	25
1.5.2	Classification des modèles existants	25
1.5.3	Les racines du mouvement.....	28
1.5.4	Un mouvement des biens communs	31
1.6	L'ingénierie des exigences.....	32
1.6.1	Principes saillants	33
1.6.2	Prototypage	35
1.6.3	Le langage univoque	36
2	Objectifs et méthodologie	39
2.1	Identification des Objectifs	40
2.1.1	Contexte.....	40
2.1.2	Protocole de recherche du projet pilote DEI	40
2.1.3	Objectifs de recherche associés à COLAB	41
2.1.4	Caractéristiques fondamentales vérifiées par COLAB	42
2.1.5	Épilogue.....	43
2.2	Contexte d'exploration.....	44
2.2.1	Parties prenantes	44
2.2.2	Sources d'exploration	44
2.3	Modèle de flux de travail	46
2.3.1	Cadre de travail	46
2.3.2	Planification.....	46
2.4	Caractérisation des productions.....	49
2.4.1	Productions logicielles (prototypes)	49
2.4.2	Productions documentaires	49
2.5	Mesures ergonomiques	51
2.5.1	Approche globale	51

2.5.2	Mesure de qualité perçue avec AttrakDiff.....	52
2.6	Méthode d'analyse	52
3	Analyse	55
3.1	Introduction.....	56
3.2	Description du domaine du problème « nu ».....	56
3.2.1	Identification des responsabilités	56
3.2.2	Pré-expérimentation	57
3.2.3	L'expérimentation	60
3.2.4	Validation d'expérimentation	63
3.2.5	L'analyse clinique	63
3.3	Plongée au cœur du problème	67
3.3.1	P0, un simple système d'information.....	68
3.3.2	P1, apparition de l'IHM	69
3.3.3	PB2, élargissement du domaine du problème et projection physique... ..	70
3.3.4	PB3, ajout d'une interface uniforme.....	71
3.3.5	PB4, essentialisation de l'environnement d'acquisition	71
3.3.6	PB5, ajout d'un modèle organisationnel.....	73
3.3.7	PA2, obligation d'un contrôle d'accès.....	74
3.3.8	PA3, intégration de l'analyse comportementale.....	75
3.3.9	PA4, exploitation des données d'analyse comportementale	76
3.3.10	PA5, interfaces de gestion.....	78
3.3.11	Synthèse des composants découverts.....	79
3.4	Modélisation organisationnelle et sociale.....	82
3.4.1	Modèle de rôles.....	82
3.4.2	Modèle coopératif	84
3.4.3	Modèle organisationnel	84
3.4.4	Modèle de collaboration	85
3.5	Modélisation de l'analyse comportementale.....	88
3.5.1	Proposition d'un modèle pour grilles d'évaluation	88
3.5.2	Recherche d'un noyau aux méta-modèles du comportement.....	91
3.5.3	Relation entre Outils d'Analyse Comportementale (OACs) et annotations	93
3.6	Modèle scénarii	94
3.7	Identification des exigences de COLAB.....	95

3.7.1	Exigences fonctionnelles	96
3.7.2	Exigences de conception et de performance	104
3.7.3	Exigences de sécurité.....	104
3.8	Caractéristiques fondamentales de API SCÉNARII	107
3.8.1	Caractéristiques de l'API SCÉNARII	107
3.8.2	Contraintes environnementales	108
4	Prototypage et validation	109
4.1	Caractérisation du prototype COLAB 0.4	110
4.1.1	Choix de technologies et déploiement.....	110
4.1.2	Considérations sur la sécurité	110
4.1.3	Fonctionnalités d'authentification	111
4.1.4	Fonctionnalités relatives à un projet	111
4.1.5	Fonctionnalités de visionnage et d'analyse d'expérimentation.....	112
4.1.6	Fonctionnalités d'administration	115
4.2	Mesures ergonomiques sur COLAB 0.4.....	116
4.2.1	Élaboration du questionnaire.....	116
4.2.2	Résultats du questionnaire	120
4.2.3	Évaluation générale et synthèse.....	128
4.3	Caractérisation du prototype medianode	129
4.3.1	Structure et déploiement avec COLAB	130
4.3.2	Aperçu du service	132
4.4	COLAB 0.5 et Serrurier, proposition d'une librairie déclarative pour automa- tiser les contrôles d'accès par méthode	133
4.4.1	Introduction à Astronomy et Meteor.....	133
4.4.2	Présentation des décorateurs en Javascript	136
4.4.3	Présentation de Serrurier	137
4.4.4	Perspectives de conception interne : Object Scope Delegate Role Ba- sed Access Control	138
5	Spécifications partielles	143
5.1	COLAB	144
5.1.1	Plan de site	144
5.1.2	Esquisses d'interface graphique	148
5.2	Coopérativisme académique des plateformes de recherches	148

5.2.1	Les déficits invisibles du milieu académique	148
5.2.2	Retour aux communs	149
Conclusion		151
A Esquisses légendées		155
A.1	Page de projet	156
A.2	Page d'étude	157
A.3	Page de configuration des outils pour étude	158
A.4	Atelier d'analyse comportementale pour expérimentation	159
A.5	Structure générale	160
B Notations		163
B.1	UML	163
B.2	Diagrammes logiques de données « Crow's Foot »	163
B.3	<i>Problem frames</i>	165
C Problem frames ou problèmes-archétype		167
C.1	Introduction à la notation des problèmes-archétype	168
C.2	Typologie des problèmes-archétype	171
C.2.1	Système-atelier	171
C.2.2	Système-contrôleur	171
C.2.3	Système d'information	172
C.2.4	Système transformationnel	175
C.2.5	Système-connecteur	175
C.3	Catégorisation d'un domaine	176
D État de l'art		179
D.1	Cotation du profil des AVQ	179
D.2	Documentation d'une tâche du profil des AVQ	180
D.2.1	Description	180
D.2.2	Compte-rendu	181
E Prototypage		185
E.1	Captures d'écran de COLAB 0.4	185
E.2	Captures d'écran de COLAB 0.5	207

E.3	Serrurier	212
E.3.1	Implémentation du décorateur « cadenas »	212
E.3.2	Implémentation du cadenas « loggedUserInRole »	213
E.3.3	Exemple de définition d'une classe avec contrôle d'accès	214
E.4	medianode	217
E.4.1	Gestionnaire d'authentification pour COLAB	217
F	Questionnaire d'utilisation de COLAB	219
F.1	Questionnaire brut.....	219
F.2	Réponses au questionnaire.....	232
	Bibliographie	249

Liste d'acronymes et d'abréviations

AC

voir Analyse Comportementale iii, xviii, xxx, 73–75, 78, 89, 99, 101, 142

AIVQ

voir Activités Instrumentales de la Vie Quotidienne 2, 13, 40, 45

ASVS

Application Security Verification Standard 102–105, 108

AVQ

voir Activités de la Vie Quotidienne xxxiv, 13, 14, 16, 89

AWM

voir Application Web Monopage xxv, 102, 108, 148

CABR

voir Contrôle d'Accès à Base de Rôle 20, 21, 24

CAD

voir Contrôle d'Accès Discretionnaire xxiv, 19, 20, 138

CAO

voir Contrôle d'Accès Obligatoire xxiv, 19, 20

DEI

Dépistage en Environnement Intelligent de la maladie d'Alzheimer xvii, xx, 2, 16, 40, 41, 44–47, 49, 54, 61, 63, 108, 109, 128, 187–191

DoD

Department of Defense 18

DOMUS

DOMotique et informatique Mobile à l'Université de Sherbrooke v, 1, 2, 44, 45, 54, 69

IHM

Interface Homme-Machine 35, 50

ILC

Interface en Ligne de Commande 77, 78, 101

LBAC

Lattice Based Access Control 19

MA

Maladie d'Alzheimer 16, 62

MTAC

voir Modèle de Tâches pour Analyse Comportementale xxxv, 73, 96, 105, 141, 149, 153–155

OAC

voir Outil d'Analyse Comportementale ix, xviii, xxxi, xxxiv, 12, 73, 78, 80, 82–86, 91, 92, 95–97, 141, 143, 148, 157

OWASP

Open Web Application Security Project 102, 105

REST

voir REpresentational State Transfer 105

TCL

Trouble Cognitif Léger 62

TSEC

Trusted Computer System Evaluation Criteria 18

Liste des tableaux

1.1	Les niveaux d'intégration à la croisée des disciplines	7
1.2	Matrice de concurrence fonctionnelle	11
1.3	Objectifs du Profil des AVQ	13
1.4	Définition des opérations dans le Profil des AVQ	17
1.5	Classification des plateformes coopératives	26
2.1	Objectifs de recherche	41
2.2	Caractéristiques fondamentales de COLAB	42
2.3	Liste des parties prenantes	43
2.4	Inventaire du matériel d'exploration et d'analyse	45
2.6	Inventaire des productions documentaires	49
2.5	Inventaire des programmes	51
3.1	Code couleur des responsabilités	57
3.2	Composants système physiques de COLAB	80
3.3	Composants système logiques de COLAB	81
3.4	Définition des responsabilités dans COLAB	82
3.5	Délégation de portée dans COLAB	83
3.6	Différents types de notifications	88
3.7	Terminologie détaillée de la grille d'évaluation	90
3.8	Comparatif	91
3.9	Liste des exigences fonctionnelles de l'application CoLab	96
3.10	Liste des exigences de conception et de performance de l'application CoLab	104
3.11	Liste des exigences de sécurité de l'application CoLab	104
4.1	Structure et exploitation du questionnaire d'utilisation de COLAB	117
4.2	Différenciateurs sémantiques du questionnaire d'utilisation	119

4.3	Exploitation de la pertinence fonctionnelle rapportée	122
4.4	Réponses « sémantisées » au questionnaire d'utilisation	122
4.5	Exploitation de la prospection fonctionnelle	126
4.6	Synthèse des retours utilisateur	129
4.7	<i>API service map</i>	133
4.8	Exemple de délégation avec des classes de COLAB 0.5	140
5.1	Plan de site pour COLAB	145
C.1	Abréviations des catégories de domaine	168
D.1	Cotation du profil des AVQ	179
F.1	Formulaire d'utilisation de COLAB	220
F.2	Réponses au formulaire d'utilisation de COLAB	233

Table des figures

1.1	Capture d'écran d'Observer XT 12.....	9
1.2	Modèle RBAC NIST 2001 et ses extensions.....	21
1.3	Hiérarchie de rôles en structure d'arbre	22
1.4	Hiérarchie de rôles en structure d'arbre inversé	23
1.5	Hiérarchie de rôles en structure de treillis	23
1.6	Origines du cooperativisme des plateformes.....	31
1.7	Vue d'ensemble sur l'ingénierie des exigences	33
1.8	Typologie des exigences	34
1.9	Modèle de processus en ingénierie des exigences	37
2.1	Étapes du projet Dépistage en Environnement Intelligent de la maladie d'Alzheimer (DEI)	47
2.2	Planification du flux de travail	48
3.1	Diagramme d'activité pré-expérimentation « nu »	59
3.2	Diagramme de contexte pour expérimentation, « nu »	61
3.3	Diagramme d'activité pour rendez-vous d'expérimentation, « nu »	62
3.4	Diagramme d'activité pour validation d'expérimentation, « nu ».....	64
3.5	Représentation sous forme d'arbre d'un patron de grille.....	65
3.6	Un cycle d'évaluation	66
3.7	Itérations des problèmes-archétypes	67
3.8	Problème-archétype P0	68
3.9	Problème-archétype P1	69
3.10	Problème-archétype PB2	70
3.11	Problème-archétype PB3	72
3.12	Problème-archétype PB4	73
3.13	Problème-archétype PB5	74

3.14 Problème-archétype PA2.....	75
3.15 Problème-archétype PA3.....	76
3.16 Projection de l'atelier d'AC sur les trois étapes d'un cycle d'évaluation.....	77
3.17 Problème-archétype PA4.....	77
3.18 Projection des systèmes transformationnels « module d'exploitation » et « module d'export ».....	78
3.19 Problème-archétype PA5.....	79
3.20 Projection des « ateliers de configuration » sur les quatre niveaux de coordination	80
3.21 Hiérarchie de rôles dans CoLab avec délégation de classes.....	84
3.22 Modèle du coopérativisme académique	85
3.23 Modèle logique organisationnel	86
3.24 Structure d'un patron de grille.....	89
3.25 Modèle de données logique des tâches.....	92
3.26 Roue typologique des relations entre OACs et annotations	94
3.27 Modèle logique de données scénarii	95
4.1 Rapport d'activité suspecte expédié par courriel à l'administrateur-système	111
4.2 Icônes de statut d'expérimentation	112
4.3 Infobulle informative sur source de captation vidéo.....	113
4.4 Infobulle informative sur les segments de tâche	114
4.5 Infobulle informative sur annotation	114
4.6 Les quatre onglets du menu contextuel de la page d'expérimentation.....	115
4.7 Diagramme des valeurs moyennes des dimensions Attrakdiff	125
4.8 Déploiement de medianode avec CoLab	131
4.9 Exemple de réponse à une requête <code>/i/:prj/:exp</code> (JSON).....	132
4.10 Déclaration d'une classe dans <code>Astronomy</code> (ECMAScript 6)	134
4.11 Isomorphisme dans Meteor avec MongoDB – Minimongo et <i>pub – sub</i>	134
4.12 Sécurisation d'une opération de mise à jour de collection dans Meteor (code serveur, ECMAScript 6)	135
4.13 Synchronisation réactive de deux sélections d'une même collection, avec DDP ...	136
4.14 Déclaration et application d'un décorateur (ECMAScript Next)	137
4.15 Utilisation du décorateur « cadenas » lors de la déclaration d'une « classe » <code>Astronomy</code> (ECMAScript Next).	138
4.16 Double contrôle d'accès client-serveur avec Astronomy et Serrurier	138
4.17 Caption	139

A.1	Esquisse de page pour projet /projects/uds/dei	156
A.2	Esquisse de page pour étude /projects/uds/dei/studies/montréal-sherbrooke	157
A.3	Esquisse de page de configuration d'outils pour étude /projects/uds/dei/studies/montréal-sherbrooke/configuration/outils	158
A.4	Esquisse d'Atelier d'AC /projects/uds/dei/studies/montréal-sherbrooke/exps/anonymous18	159
A.5	Structure générale d'une page de l'application COLAB.....	160
A.6	Accès rapides dans l'application COLAB.....	161
A.7	Notifications dans l'application COLAB	162
B.1	Légende de cardinalité dans les diagrammes logiques de données « Crow's Foot »	164
B.2	Diagramme logique de données « Crow's Foot », exemple simplifié	164
B.3	Légende des diagrammes <i>problem frames</i>	165
C.1	Problème-archétype élémentaire, un logiciel de dessin vectoriel	169
C.2	Problème-archétype composite, un logiciel de dessin vectoriel	169
C.3	Problème-archétype : système-atelier	172
C.4	Problème-archétype : système-contrôleur, variante automatique	172
C.5	Problème-archétype : système-contrôleur, variante commandée	172
C.6	Problème-archétype : système d'information, variante temps-réel	173
C.7	Problème-archétype : système d'information, variante sur demande	174
C.8	Problème-archétype : système d'information, variante sur demande avec dispositif d'actualisation	174
C.9	Problème-archétype : système d'information, variante sur demande simplifiée	174
C.10	Problème-archétype : système transformationnel.....	176
C.11	Problème-archétype : système-connecteur, variante excentrée	176
C.12	Problème-archétype : système-connecteur, variante centrée	176
C.13	Catégorisation d'un domaine	177
D.1	Documentation de la tâche « Téléphoner pour une information » du profil des AVQ	180
E.1	Capture d'écran de la page d'authentification /user/login	186
E.2	Capture d'écran de la page de création de compte /user/register, visuel par défaut	187

E.3	Capture d'écran de la page de création de compte /user/register, mot de passe invalide et adresse courriel absente de la liste blanche	188
E.4	Capture d'écran de la page de tableau de bord /user/dashboard	189
E.5	Capture d'écran de la page d'intégration aux projets /user/join-project	190
E.6	Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un associé.....	191
E.7	Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un associé, avec saisie d'un filtre textuel.....	192
E.8	Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un responsable.....	193
E.9	Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un associé, avec saisie d'un filtre textuel.....	194
E.10	Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur le panneau de configuration par un responsable	195
E.11	Capture d'écran de la page de gestions des membres d'un projet /projects/DEI /users, une demande d'intégration au projet est en suspend	196
E.12	Capture d'écran de la page de configuration des règles d'un projet /projects /DEI/rules	197
E.13	Capture d'écran de la page de définition d'un <i>template</i> de projet /projects /DEI/template, visuel sur l'arborescence de tâches.....	198
E.14	Capture d'écran de la page de définition d'un <i>template</i> de projet /projects /DEI/template, visuel sur l'arborescence d'annotations	199
E.15	Capture d'écran de la page d'expérimentation /projects/DEI/exps/56a6982e4c7 cbf05907982f7 , visuel par défaut	200
E.16	Capture d'écran de la page d'expérimentation /projects/DEI/exps/56a6982e4c7 cbf05907982f7, visuel par défaut	201
E.17	Capture d'écran de la page d'expérimentation, visuel étendu.....	202
E.18	Capture d'écran de la page d'administration des règles /admin/rules	203
E.19	Capture d'écran de la page d'administration des utilisateurs /admin/users, visuel sur la liste blanche	204
E.20	Capture d'écran de la page d'administration des utilisateurs /admin/users, visuel sur la liste d'utilisateurs	205
E.21	Capture d'écran de la page d'administration du serveur vidéo /admin/videos ..	206
E.22	Capture d'écran de la page d'authentification /1.....	208

E.23 Capture d'écran de la page de tableau de bord /d	209
E.24 Capture d'écran de la page relative à l'expérimentation EXP26 du projet DEI /e/DEI/EXP20, sans explorateur temporel	210
E.25 Capture d'écran de la page relative à l'expérimentation EXP26 du projet DEI /e/DEI/EXP20, avec explorateur temporel	211

Glossaire technique

application web monopage

Génie logiciel. Application web qui n'est chargée qu'une seule fois, c'est-à-dire que la navigation entre différentes pages de l'application ne nécessite pas le rechargement de l'ensemble des ressources (HTML, CSS, JavaScript) par le navigateur. [xiii](#), [xxv](#), [108](#), [148](#)

atelier

Génie logiciel. Un système-atelier est défini par JACKSON comme un outil permettant à un utilisateur de créer et d'éditer un certain type de texte, de données structurées ou d'objets graphiques, de sorte qu'ils puissent être copiés, imprimés, analysés ou utilisés de toute autre manière [[40](#), page 77]. [xxix](#)

bash

Logiciel. Acronyme de *Bourne-Again shell*. C'est l'interpréteur en ligne de commande de type script le plus populaire dans les systèmes d'exploitation Unix, dont GNU/Linux et Mac OS X. [49](#)

chaînes de blocs

Protocole. De l'anglais *blockchain*. Les chaînes de bloc peuvent être vues comme des bases de données distribuées et transparentes, sans élément central de contrôle. Un seul « univers » est couvert par cette base de données, qui conserve l'historique de toutes les transactions effectuées depuis son lancement par tous les agents à travers le monde ayant créé un compte (ou portefeuille). L'intégrité est garantie par un système de consensus entre chaque agent du réseau distribué. [xxxii](#), [27](#), [28](#), [32](#)

contrôle d'accès à base de rôle

Génie logiciel. De l'anglais *Role Based Access Control*. Se dit d'une politique de contrôle d'accès fondée sur une logique de rôles. L'idée est de transposer la place de chaque individu dans la société dans une structure de rôles qui contraint les accès aux ressources détenues par l'entreprise ou l'organisme. [xiii](#), [20](#), [21](#), [24](#)

contrôle d'accès discrétionnaire

Génie logiciel. De l'anglais *Discretionary Access Control*. Se dit d'une politique de contrôle d'accès où les utilisateurs sont propriétaires des ressources auxquelles ils accèdent, et peuvent donc pas déléguer leurs accès à d'autres utilisateurs. S'oppose aux CAO. xiii, xxiv, 19, 20

contrôle d'accès obligatoire

Génie logiciel. De l'anglais *Mandatory Access Control*. Se dit d'une politique de contrôle d'accès où les utilisateurs ne sont pas propriétaires des ressources auxquelles ils accèdent, et ne peuvent donc pas déléguer leur accès. L'organisme ou l'entreprise est propriétaire de la ressource, et est donc seul garant de leur exposition. S'oppose aux CAD. xiii, xxiv, 19, 20

domaine d'application

Ingénierie des exigences. Le domaine d'application (*application domain*) est un cas particulier du domaine du problème lorsque le système-solution est purement logiciel [7, page 9]. 5, 18, 20, 33, 35, 44, 47

domaine du problème

Ingénierie des exigences. Le domaine du problème (*problem domain*) est « la portion de l'Univers dans laquelle le problème existe » [7, page 9]. C'est à ce problème que va répondre la solution logicielle. xxiv, xxv, xxix, 3, 33, 34, 36, 45, 48, 51, 54, 65, 163, 167

domaine métier

Le domaine métier, c'est la partie du domaine du problème qui concerne une expertise dont il faudra tirer les connaissances, les protocoles, les savoirs-faire et les pratiques avant de concevoir le système-solution. Ce domaine est généralement intangible. 36, 44, 45, 65, 115

environnement opératoire

Ingénierie des exigences. L'environnement opératoire (*operating environment* [7, page 9]) est l'espace tangible du réel qui s'interface avec le système-solution. 44, 45, 54, 65, 69

full stack

Technologie. Dans un système à trois couches, se dit d'un cadriceil traversant ces trois couches directement, en offrant une interface de programmation homogène pour le développeur. Ces niveaux sont « présentation », « logique métier » et « sources de données ». 108

intégrité

Sciences de l'information. L'intégrité est un concept abstrait qui, dans le contexte des technologies de l'information, a deux significations fortement liées [50, pages 16-17]. Dans un système d'information, l'intégrité des données désigne la préservation des informations au cours du temps. Plus précisément, la préservation de leur qualité, leur authenticité et leur exactitude. Par extension, il désigne la capacité de résilience d'un système d'information face à des épisodes de corruption, intentionnels ou non. On parle alors d'intégrité du système pour désigner l'application constante des règles définies par le système qui conditionne l'intégrité des données. L'intégrité du système est souvent associée au domaine de la tolérance à l'erreur (*domain fault tolerance*). 18

isomorphisme

Web. Il désigne, dans une application web, une propriété offerte par le partage d'un même langage de programmation entre le programme exécuté sur le serveur et celui exécuté sur le client, en l'occurrence : Javascript. Certaines sources sont partagées entre les deux couches, présentation et métier. Cette mise en commun offre l'avantage d'avoir une représentation unifiée de la logique métier, réduisant potentiellement le coût d'une évolution du modèle du domaine et accélérant le développement. 108, 132, 133, 135

langage omniprésent

Génie logiciel. Le langage omniprésent ou univoque (*ubiquitous language*) est popularisé par EVANS en 2003 [19]. Le glossaire univoque qu'il prescrit doit être partagé entre développeurs, architectes, coordinateurs, clients et responsables de produit pour fluidifier la communication et homogénéiser le vocabulaire par raffinements successifs. Plus qu'une simple appropriation du langage expert par l'équipe de travail, il incite aussi à dépouiller ce dernier de toutes ses ambiguïtés, contradictions logiques et autres chevauchements sémantiques, et ne constitue donc pas un processus neutre vis-à-vis du domaine du problème. xxix, 115

Meteor

Cadriciel pour AWM réactive [52]. 108, 131, 133

modèle analogique

Ingénierie des exigences. Représentation interne d'une portion du réel par un système d'information. Un tel modèle peut prendre la forme d'une ontologie, d'une base de données relationnelle, d'un modèle de classes objet, ... etc [40, page 13]. Distinct du modèle analytique. xxvi, xxix, 9, 36, 86, 174

modèle analytique

Ingénierie des exigences. Description d'un processus ou d'une portion du réel comprenant des lois, des relations, des équations et autres propriétés. Cette description est une abstraction, c'est-à-dire une simplification de la réalité par extractions de caractéristiques générales. Elle vise à rendre intelligibles aux humains des phénomènes complexes, difficiles à appréhender [40, page 13]. Distinct du **modèle analogique**. xxv, 36

prototype exploratoire

Ingénierie des exigences. Prototype utilisé pour l'exploration et la définition d'exigences (analyse). Ce prototype est abandonné une fois la phase de spécification entamée [7, page 242]. 35, 49

prototype structurel

Ingénierie des exigences. Prototype dont l'utilité est d'évaluer des pistes de conception interne pour valider, par exemple, des contraintes de performance. Il peut aussi être utilisé dans la phase d'exigences pour évaluer la faisabilité et la pertinence du système-solution [7, page 242]. 35, 46, 49

qualité hédonique

Ergonomie. La qualité hédonique (perçue) est non instrumentale et se réfère au soi. Elle est donc liée à l'utilisateur, et se base sur un jugement du potentiel du produit à procurer du plaisir et à satisfaire l'épanouissement de besoins humains plus profonds appelés « be-goals ». La capacité du système interactif à stimuler l'utilisateur, à le connecter aux autres, à lui donner un sentiment de contrôle ou encore à lui conférer une certaine popularité sont des attributs liés à la qualité hédonique [47]. 50, 51

qualité pragmatique

Ergonomie. La qualité pragmatique (perçue) désigne les aspects instrumentaux du système ou produit, c'est-à-dire son utilité et son utilisabilité. Ces qualités pragmatiques vont soutenir la réalisation d'objectifs-tâches (appelés « do-goals »). La clarté du système, sa structure et sa prévisibilité sont autant d'attributs liés à la qualité pragmatique [47]. 51

réactive

Génie logiciel. Se dit d'une application qui actualise les données présentées à l'utilisateur à mesure qu'elles sont modifiées dans le modèle persisté. Autrement dit, l'état interne de l'application cliente est toujours synchronisé — ou en cours de synchronisation — avec le

modèle de données persisté, à l'inverse d'une application web classique où l'actualisation se fait à la demande de l'utilisateur, lorsqu'il navigue par hyperliens. xxv, 108, 133

representational state transfer

Programmation. Style architectural qui profite des méthodes et propriétés du protocole HTTP pour faire de l'interface de programmation un domaine navigable de manière analogue à la navigation par hyperliens dans un navigateur web. xiv

vlc

Logiciel. Lecteur vidéo populaire, ses fonctionnalités avancées permettent l'acquisition de flux vidéo réseau RTSP. Il peut aussi être utilisé en ligne de commande. 49, 145

Dictionnaire de domaine univoque

*Ce dictionnaire vise à clarifier les termes qui sortent des champs de l'ingénierie logicielle et de l'informatique. Sa présence est justifiée par la méthode du langage **omniprésent**, proposée par EVANS et détaillée dans la **sous-section 1.6.3**. Certains concepts sont internes au **modèle analogique**, et ces termes sont précédés de la mention CoLab puis éventuellement suffixés d'un artefact (par exemple, CoLab:Responsabilité). Il est important de noter que puisque COLAB est en grande partie un **outil-atelier**, une portion non-négligeable du **domaine du problème** est réifié dans l'application, c'est-à-dire qu'elle n'a pas de sens en dehors de l'application, ou bien un sens dégradé et flou. Pour illustrer ce constat, on peut prendre l'exemple de la responsabilité « **associé d'étude** ». Dans COLAB, cette responsabilité est associée à un ensemble de permissions qui font partie du contrôle d'accès, et les fonctionnalités associées sont relatives aux objectifs-tâches de l'utilisateur. On ne peut pas se contenter ici de faire une transposition mécanique entre le modèle analytique et le modèle analogique, puisque l'application crée un nouveau domaine de travail dans le cyberspace.*

accord interjuge

Ergothérapie, CoLab. L'accord interjuge désigne, dans le cadre d'une **cotation de grille**, la tendance de chaque observateur à identifier les mêmes critères de grille que son homologue. On parle de « **fidélité interjuge** » pour mesurer l'homogénéité ou la dispersion entre les évaluations de chaque juge. 62, 63, 84, 85, 96

activités de la vie quotidienne

Ergothérapie. HAMONET et BÉGUÉ-SIMON : « Des activités routinières effectuées par un individu à chaque jour ou de façon habituelle dans divers environnements. Ce sont des activités propres à donner à la personne une autonomie individuelle assurant sa survie personnelle et son maintien dans la communauté. Certaines activités sont indispensables

à la survie de l'individu et d'autres sont davantage liées à la socialisation. » (31) xiii, xxxiv, 13, 14, 16, 89

activités instrumentales de la vie quotidienne

Ergothérapie. Les activités instrumentales de la vie quotidienne (AIVQ) sont des composantes du comportement habituel des humains qui ne sont pas nécessaires à leur fonctionnement de base (par opposition aux activités de la vie quotidienne), mais qui permettent à chacun de vivre de manière autonome dans sa communauté. xiii, 13, 40, 45

administrateur d'organisme

CoLab:Responsabilité. Un administrateur d'organisme est en charge de la configuration de l'API SCÉNARII. Il choisit aussi la politique d'inscription des projets sous la responsabilité de l'organisme. Il a le pouvoir de restreindre l'accès et supprimer des données d'expérimentation relatives à l'API SCÉNARII dont il est responsable. 80, 85, 94, 100, 105, 143

administrateur global

CoLab:Responsabilité. Un administrateur possède un contrôle total sur l'application, à l'exception des études. Il est notamment en charge d'autoriser la configuration de nouveaux organismes. 80, 101, 109, 113

analyse comportementale

L'analyse comportementale désigne l'ensemble des pratiques consistant à étudier le comportement d'un ou plusieurs individus. Ce terme relativement générique recouvre des pratiques issues de disciplines comme l'éthologie, la psychologie, l'ergothérapie ou encore la neurologie. iii, xiii, xviii, xxx, 73–75, 78, 89, 99, 101, 142

analyste comportemental

CoLab. C'est un collaborateur qui participe à l'AC. xxxiii–xxxv, 40, 61, 63, 68, 73, 74, 86, 97–99, 149

annotation

CoLab. Terme générique qui désigne toute donnée produite par un agent relativement à une expérimentation. On distingue alors les annotations, produites par des agents à partir de données d'acquisition, de ces dernières, extraites par des machines dans l'environnement d'acquisition. Parmi ces données annotées, on peut citer :

- les tâches réalisées ;
- les formulaires de grille ;

— les observation de verbatim.

xxxi, xxxiii, xxxv, xxxviii, 74, 80, 81, 84–86, 91, 94, 95, 98, 142, 147

associé d'étude

CoLab:Responsabilité. Un associé d'étude est un utilisateur qui a été intégré dans une étude. Il peut notamment produire des annotations à partir des OAC déployés pour l'étude.

xxix, xxxiii, xxxviii, 80, 81, 83, 85, 94–96, 99–101, 142, 143, 147, 153, 158

catégorie d'observation

CoLab. Les types d'observation de grille peuvent être regroupées en catégories dans une structure d'arbre distincte du patron de grille. Cette structure permet de regrouper des observations atomiques dans une typologie. Par exemple, on pourrait avoir la catégorie « erreur » qui comprend les observations de type « omission », « égarement », « substitution », « addition » et « abandon ». À noter qu'une observation peut avoir plusieurs types.

xxxv, 73, 85, 88, 89, 91, 96, 141

chargé de recrutement

API Scénarii:Responsabilité. Le chargé de recrutement doit trouver des participants pour les expérimentations à venir. 55, 58

classe du participant

Statistique, CoLab. La classe du participant est une variable modale. C'est une catégorie de diagnostic *a priori* du participant. Par exemple, « trouble cognitif léger », « sain », « maladie d'Alzheimer ».

xxxiii, 55, 61, 62, 110

collaborateur

CoLab:Responsabilité. Un collaborateur est un utilisateur qui a été intégré à un projet. Il dispose alors de droits d'accès aux données d'expérimentation pour les projets dans lesquels il est collaborateur. Il peut aussi y créer des études. xxx, xxxiii, xxxvi, xxxvii, 55, 61, 66, 73, 74, 76, 80, 81, 85, 94–97, 100, 110, 111, 142, 158

conditions initiales d'expérimentation

En respect de la méthode scientifique, et afin d'isoler et contrôler les paramètres d'expérimentation, c'est l'ensemble des prérequis pour garantir que, d'une expérimentation à l'autre :

— les participants évoluent dans un environnement fixe ;

- les données sont récoltées à partir de capteurs dont le positionnement et la configuration demeurent constants.

On distingue les conditions initiales techniques, qui sont relatives au système d'acquisition de données, et les conditions initiales environnementales, relatives à l'état général de l'appartement, la disposition des objets et la propreté. xxxv, 55, 56, 58, 149

cotation de grille

Ergothérapie, CoLab. Application du **patron de grille** à une **expérimentation** particulière par l'intermédiaire du **formulaire de grille**. Une analogie peut être faite avec un questionnaire, qui serait le patron, et les réponses au questionnaire, qui seraient les **formulaires de grille**. xxix, xxxiv, 15, 61, 62, 84, 85, 88

critère de grille

CoLab. Un critère de grille est une feuille du **patron de grille** qui va correspondre, pour une **expérimentation** donnée, à une saisie par l'évaluateur. Un critère de grille est toujours typé, c'est-à-dire qu'il contraint les valeurs possibles de la saisie (par ex., texte, valeurs modales). xxxiv, xxxvi, 61, 86, 88

désintermédiation

Économie. En économie, une transaction désintermédiatisée est effectuée sans le concours d'une tierce partie, d'un « médiateur » qui, en général, en tire une commission. On peut songer, par exemple, à une transaction par **chaînes de blocs** qui ne nécessite ni banque, ni organisme central pour confirmer et garantir l'intégrité de la transaction, même si des frais infinitésimaux peuvent s'appliquer. 1

économie du partage

Économie. L'économie du partage désigne l'ensemble des transactions — pas nécessairement marchandes — qui sont permises par le partage de biens entre pairs. Le covoiturage, les services de chauffeurs, les mutualisation des machines dans les *fablabs*, la location de biens immobiliers en ligne ou encore le partage d'outils en ligne participent à cette économie. 1, 28, 29

ergothérapie

Profession. L'ergothérapie est une profession paramédicale évaluant et traitant les personnes afin de préserver et développer leur indépendance et leur autonomie dans leur environnement quotidien et social. 14, 45

étude

CoLab. Une étude réunit un ou plusieurs **collaborateurs** pour faire une évaluation à partir des données d'expérimentation. Le collaborateur qui a l'initiative de l'étude en est le **gestionnaire**. Il peut y inviter d'autres collaborateurs. Il peut rendre l'étude ouverte aux autres collaborateurs. **xxx, xxxi, xxxiv, xxxvi, 80, 81, 83, 85, 94–101, 141–143, 148, 157**

évaluation réflexive

CoLab. Une évaluation réflexive ou évaluation de confiance est une note allant de 0 à 5. Cette note porte sur la confiance que l'**analyste comportemental** a dans une **annotation** qu'il a lui-même produite. S'il doute fortement, il peut mettre une note assez faible qui ressortira dans des étapes de collaboration ultérieures, et pourra aisément être discutée avec les autres **associés d'étude**. **12, 84, 95, 98, 147**

examineur-accompagnateur

API Scénarii:Responsabilité. Il est chargé d'accompagner et superviser le **participant** avant et durant l'**expérimentation**. La nature de son assistance et les règles d'intervention sont déterminées par le protocole de recherche du projet associé. **xxxv, 55, 58, 61**

expérimentation

CoLab. Dans le cadre d'un projet de recherche, activité supervisée et enregistrée par un environnement d'acquisition dans laquelle le sujet d'étude est un **participant**. L'activité doit se dérouler à portée de l'environnement d'acquisition, pour qu'elle puisse être analysée *a posteriori*. **xxxi–xxxiv, xxxvi, xxxviii, 10, 61, 62, 66, 67, 69, 74, 81, 82, 84–86, 88, 94, 97, 99, 110, 111, 127, 142, 147, 158**

expérimentation floutée

CoLab. Expérimentation dont certaines informations sont cachées au **collaborateur**. Ces informations pourraient biaiser son évaluation en compromettant l'« aveugle ». Une expérimentation floutée est identifiable par un numéro aléatoire qui ne suit pas l'ordre chronologique des expérimentations. Cet ordre pourrait en effet contenir des motifs relatifs aux **classes des participants**. On parle de « lever le floutage » lorsque ces informations sont révélées au **collaborateur**. **12, 83, 84, 95, 100, 110, 142, 148**

explorateur temporel

CoLab. L'explorateur temporel est un élément de l'interface graphique qui permet de visualiser en localisant dans le temps différentes données d'expérimentation, comme des activités, de la détection de mouvement, des événements de capteurs ou encore des observations de grille et de verbatim. **xxxvii, 87, 97, 98, 155**

fonctions exécutives

Sciences cognitives. C'est la capacité à planifier et réaliser des comportements complexes orientés vers des buts. Elle comprend notamment des aptitudes à formuler des buts, planifier et réaliser des tâches puis enfin, s'assurer que les buts visés ont été atteints. 16

formulaire de grille

CoLab. Le formulaire de grille est dérivé du *patron de grille* et permet de saisir des *critères de grille* pour une *expérimentation* en particulier. L'activité qui consiste à saisir ce formulaire est appelée *cotation de grille*. xxx, xxxii, xxxiv, xxxvi, 74, 86, 88, 101

gestionnaire d'étude

CoLab:Responsabilité. Il est en charge d'une *étude*. Il doit notamment choisir les outils d'analyse utilisés par le groupe d'étude, fixer des jalons et assurer la coordination entre les associés. xxxiii, 80, 81, 83–85, 94–96, 143, 153, 154

grille d'évaluation

CoLab, Ergothérapie. C'est un outil générique qui permet aux *analystes comportementaux* d'évaluer le comportement des *participants* à partir de critères objectifs qu'ils ont pu définir. On distingue plus spécifiquement :

- le *patron de grille*, qui normalise l'évaluation ;
- un *formulaire de grille*, qui est dérivé du *patron de grille*. Il correspond aux saisies de critères de grille pour une *expérimentation* particulière.

xxxv, 15, 61, 86, 99, 101, 123, 141, 149, 154, 155

magasin libre des universités

CoLab. Espace où les *utilisateurs* peuvent consulter et importer des *OAC* suivant une license choisie par leurs auteurs. 83, 95–97, 99, 141, 154, 157

modèle de tâches pour analyse comportementale

Analyse Comportementale/CoLab. Un modèle de tâches pour Analyse Comportementale (MTAC) décrit une hiérarchie de tâches relativement à un protocole de recherche. Par exemple, le profil des *AVQs (AVQs)* propose vingt tâches, chacune décomposée en sous-tâches. On pourrait alors définir un modèle de tâches du profil des *AVQs*. xiv, xxxv, 73, 96, 105, 141, 153

observation de grille

CoLab. Observation atomique, c'est-à-dire localisée dans le temps. Elle est rattachée à un *critère de grille* dont le type est un ensemble d'observations. Les types d'observation

peuvent être définis dans des catégories d'observation. xxxi, xxxvii, 74, 85, 87, 88, 91, 98, 99, 155

observation de verbatim

Ergothérapie, CoLab. Description textuelle, concise et horodatée rapportant un geste, une mimique, une vocalisation ou une interaction d'un sujet observé. xxxi, xxxvii, xxxviii, 74, 91, 98, 99, 155

opérateur d'expérimentation

API Scénarii:Responsabilité. Il est en charge du respect des conditions initiales d'expérimentation et peut accessoirement assister l'examineur-accompagnateur. 55, 68, 149

opérateur en domotique

CoLab:Responsabilité, API Scénarii:Responsabilité. Un opérateur domotique est en charge du bon fonctionnement des dispositifs d'acquisition du laboratoire où ont lieu les expérimentations (le site). Il valide l'expérimentation d'un point de vue technique, et documente des erreurs d'acquisition si nécessaire. 40, 55, 58, 61, 80, 85, 100, 105, 106, 111, 112, 142, 149

organisme

CoLab. Un organisme correspond à l'entité légale en charge d'une API SCÉNARII. Elle a aussi la charge d'un projet, et de protéger la confidentialité des données de participants. xxx, xxxvi, 80, 81, 94, 101, 105, 143, 146

outil d'analyse comportementale

Les Outils d'Analyse Comportementale (OAC) assistent l'analyste dans l'évaluation de participants. Pour le moment, trois types d'outils ont été identifiés :

- les grilles d'évaluation ;
- le verbatim ;
- les Modèles de Tâches pour Analyse Comportementale (MTACs).

Les grilles et les MTACs sont paramétrés par l'utilisateur qui en crée des instances, à la manière d'un forgeron qui élabore ses propres outils. Le paramétrage de ces outils va contraindre la structure et les valeurs des annotations qui pourront être produites pour l'évaluation. COLAB prévoit la possibilité pour la communauté de créer de nouveaux types d'outils, afin d'étendre les possibilités offertes par l'application. ix, xiv, xxxi, xxxiv, 78, 80, 82, 84, 85, 95–97, 141, 157

participant

Un participant est un sujet d'expérimentation. xxxiii–xxxv, xxxvii, 58, 61, 86, 111, 112, 127

patron de grille

CoLab. Ensemble de critères structurés sous forme d'arbre dont les feuilles sont des critères de grille, et dont les nœuds de niveaux inférieurs constituent des catégories d'autant plus générales qu'elles sont proches de la racine. Les nœuds servent donc à structurer et abstraire la grille tandis que les feuilles permettent de dériver le formulaire de grille. xxxi, xxxii, xxxiv, 61, 73, 84, 86, 88, 96

projet

CoLab. Un projet est un travail de recherche collaboratif nécessitant la coopération de chercheurs, étudiants et professionnels à travers un ou plusieurs établissements (organismes). À un projet est associé une liste d'expérimentations, un ou plusieurs sites où ont lieu ces dernières, un ensemble de collaborateurs et enfin des études. xxxi, 66, 73, 80, 81, 83, 85, 94–96, 99, 100, 105, 128, 141, 142, 148, 149, 157

réflexivité

Épistémologie. La réflexivité en sciences, c'est la capacité de regard et d'introspection sur sa propre démarche scientifique, voire l'étude de sa propre méthode. Le concept a été introduit par Pierre BOUDIEU. Il a notamment étudié les mécanismes sociaux à l'oeuvre dans la « sélection naturelle » appliquée aux théories scientifiques. 42, 147, 148

responsabilité

CoLab. À partir de la variation n°2 de VINCENT, la responsabilité est un ensemble d'objectifs associés à la fonction d'un acteur vis-à-vis du projet.

Philosophie:Morale. Dans sa taxonomie, elle distingue six variations autour du concept de responsabilité [72] :

1. (responsabilité de vertu) qualité d'une personne qui agit consciencieusement, et qui tire de cette attitude une certaine réputation ;
2. (responsabilité de rôle) *devoir* d'une personne qui a une charge, souvent associée à un titre ;
3. (responsabilité des conséquences) devoir moral de l'individu qui doit assumer une erreur ;
4. (responsabilité causale) responsabilité est synonyme de « qui est à l'origine de » ;

5. (responsabilité de capacité) on dit qu'un individu est « responsable » pour souligner qu'il n'a pas d'« excuse » qui aurait pu réduire son jugement et sa capacité à faire des choix judicieux ;
6. (responsabilité légale) attribution d'une faute pénale et obligation d'en « payer » le tribut.

54, 109, 111, 158

responsable de projet

CoLab:Responsabilité. Un responsable de projet a la charge de la gestion des collaborateurs du projet. 80, 94, 96, 110–112, 142, 152

scénario d'acquisition

CoLab/API Scénarii. Un scénario d'acquisition correspond à une séquence temporelle horodatée pendant laquelle sont enregistrées des données d'expérimentation dans un environnement d'expérimentation. Un tel scénario se décompose en trois phases :

- amorçage, lorsque l'acquisition démarre mais que l'expérimentation n'a pas encore démarré ;
- expérimentation, lorsque l'expérimentation se déroule.

Il y a donc trois événements assimilables à un scénarii d'acquisition. Le « début d'acquisition », le « début d'expérimentation » et enfin la « fin d'expérimentation » 92, 105, 106, 127

signet

CoLab. Un signet est un marqueur de temps visible dans l'explorateur temporel. Il peut être converti en observation de verbatim, en observation de grille ou encore en début de segment de tâche. 98

tâche réalisée

Analyse Comportementale/CoLab. Tâche effectuée par un participant. xxx, 74, 92

utilisateur

CoLab. Un individu inscrit sur COLAB. xxxi, xxxiv, xxxv, xxxviii, 80, 81, 85, 94, 96, 97, 100, 101, 140, 141, 143, 158

verbatim

Ergothérapie, CoLab. Le verbatim est la retranscription textuelle de paroles, et éventuellement d'actes, de gestes et d'expressions corporelles, constituée d'un ensemble d'observations horodatées. xxxv, 119, 123

visiteur d'étude

CoLab:Responsabilité. Un visiteur d'étude est un utilisateur qui a des permissions dégradées par rapport à l'associé d'étude. Il peut consulter les expérimentations, les annotations et les résultats, mais en lecture seule, sans possibilité d'édition. Il a cependant le droit de participer aux discussions de l'espace prévu à cet effet. 81, 84, 95, 96, 147

Introduction

1 Contexte global

Le monde contemporain connaît des bouleversements incessants liés à la disruption de nouvelles technologies dans presque toutes les facettes de notre quotidien [64]. Les transformations associées à l’essor du web affectent entre autres les modalités de la coopération, avec l’émergence de nombreux outils destinés à la collaboration à distance. La **désintermédiation** progressive de l’économie réduit les obstacles à la coopération et la collaboration spontanée. Mais cette désintermédiation a été freinée par l’arrivée de nouveaux monopoles [46], ce qui a favorisé à partir de 2014 l’émergence du « coopérativisme des plateformes » qui promeut la réappropriation par la société civile des territoires colonisés et monopolisés de l’**économie du partage** [68]. Ces évolutions rapides affectent naturellement le monde de la recherche, avec une intensité qui diffère suivant les champs disciplinaires, les résistances générationnelles, académiques et les cadres culturels.

De nombreux logiciels offrent aujourd’hui des fonctionnalités de collaboration. En premier lieu, les éditeurs de texte à l’instar de Google docs ou Sharelatex. Puis, des outils d’annotation de ressources, comme le logiciel libre Hypothes.is [60]. Enfin, les outils liés à la gestion d’équipes et de projets connaissent un grand succès, portés par la promesse d’un *management* facilité.

Il apparaît que le domaine couvert par ces logiciels est relativement restreint compte tenu des possibilités infinies offertes par la collaboration en ligne. De nouvelles innovations facilitant le partage et l’exploitation des données de recherche ont un fort potentiel d’améliorer la coordination dans des équipes présentant notamment les caractéristiques suivantes :

- pluridisciplinaires ;
- éclatées entre différents établissements voire pays.

Le logiciel COLAB, issu de travaux réalisés dans le **DOMotique et informatique Mobile** à l’Université de Sherbrooke (DOMUS), s’inscrit alors dans cette perspective.

2 Contexte local

Le laboratoire **DOMUS** s'est associé à l'Université de Montréal pour évaluer la faisabilité d'un dépistage précoce d'Alzheimer en appartement « intelligent ». Le projet **Dépistage en Environnement Intelligent de la maladie d'Alzheimer (DEI)** a nécessité la tenue d'expérimentations pendant lesquelles des participants devaient réaliser des **Activités Instrumentales de la Vie Quotidienne (AIVQ)**. Ces expérimentations ont été enregistrées par différents dispositifs (vidéo, audio, capteurs) pour permettre d'une part l'analyse *a posteriori*, d'autre part une recherche de corrélations éventuelles entre les données récoltées et le diagnostic des participants. Pendant ces expérimentations, un opérateur accompagnait l'évaluateur pour saisir, dans une application sur tablette, les activités effectuées par chaque participant.

La double contrainte de respecter la confidentialité des participants et de permettre aux analystes d'accéder aux vidéos à distance — ajoutée au bénéfice de disposer des activités saisies par l'opérateur d'expérimentation — ont convaincu de l'intérêt d'apporter une solution logicielle sécurisée. Les objectifs de recherche sont multiples : on veut évaluer comment la solution peut être conçue pour « permettre et encourager la collaboration de différentes expertises autour d'un projet d'étude » (interdisciplinarité, **O.I**), mais aussi dans quelle mesure elle peut être rendue générique eu égard de la diversité des disciplines mobilisées dans la collaboration (généricité, **O.II**). Aussi, on souhaite « identifier les caractéristiques que doit satisfaire l'environnement d'acquisition » pour étendre la compatibilité du dispositif avec un maximum de laboratoires partenaires (adaptabilité, **O.III**). Enfin, on souhaite réfléchir à la façon dont l'outil peut, une fois implémenté, « progresser et perdurer dans le milieu académique » en garantissant la protection des données confidentielles des participants (gouvernance, **O.IV**).

Le présent travail s'attelle donc à décrire avec autant de détails que possible les caractéristiques de la solution pour qu'elle réponde à tous ces objectifs. En plus de caractéristiques techniques et métier, une piste de gouvernance directement inspirée du coopérativisme des plateformes est envisagée. Autant pour répondre aux difficultés à maintenir des outils produits par le milieu académique que pour renforcer la valeur centrale qu'il doit attribuer aux biens communs et au partage de la connaissance.

Dans un premier chapitre (**chapitre 1**), un état de l'art « éclectique » sera fait, à l'image du problème protéiforme abordé par le présent mémoire. Ensuite seront présentés la méthodologie, ancrée dans l'ingénierie des exigences, et les objectifs de recherche (**chapitre 2**). Puis, trois chapitres présenteront les fruits du travail accompli pour répondre au problème posé. Dans la tradition de l'ingénierie des exigences, une première étape va être d'analyser en profondeur le

domaine du problème (chapitre 3). En parallèle, des pistes de solution sont testées (chapitre 4). Enfin, la synthèse de ces deux parties aboutira à la présentation de spécifications partielles (chapitre 5). La conclusion présentera les limites de ce travail et les perspectives qu'il ouvre.

Chapitre 1

État de l’art

En ingénierie des exigences, le domaine d’application est la portion du monde réel concernée par le ou les problèmes que la solution envisagée vise à élucider. Ce chapitre a pour but de rendre compte, d’une part, des travaux nécessaires à la compréhension du présent mémoire, et d’autre part, à préparer le terrain pour l’analyse du domaine d’application. Dans la [section 1.1](#), on approche l’interdisciplinarité et ses concepts dérivés. Puis, dans la [section 1.2](#), on fait l’inventaire des logiciels d’analyse comportementale concurrents. Ensuite, la description d’une méthode d’analyse comportementale intitulée « profil des AVQ » est faite ([section 1.3](#)). Dans la [section 1.4](#), un aperçu des contrôles d’accès est proposé. Une présentation du coopérativisme des plateformes est offerte dans la [section 1.5](#). Enfin, la [section 1.6](#) présente la démarche générale de l’ingénierie des exigences.

1.1 Pluri, inter et transdisciplinarité

Avant de poser le contexte de ce travail, il est nécessaire de définir le cadre sémantique. L'interdisciplinarité est aujourd'hui souvent utilisée de manière revendicative, comme l'affirmation d'une volonté d'ouverture des disciplines les unes aux autres, mais cette revendication ne garantit en rien qu'elle se réalise. Cette partie vise à approcher la nature de l'interdisciplinarité, ainsi que ses concepts voisins mais distincts. La notion de transdisciplinarité a été inventée par Jean PIAGET en 1970 pour dépasser le cloisonnement des sciences inhérent à la complexification des domaines. KLEIN propose une taxinomie de l'interdisciplinarité. Elle propose différents axes discriminatoires. L'un est le degré de collaboration ou d'interaction disciplinaire, l'autre le niveau d'intégration. CHARAUDEAU propose ainsi une distinction entre ces différents concepts :

« dorénavant, aucune discipline – hors le moment de sa procédure d'analyse – ne peut rester enfermée dans son orthodoxie. Elle doit accepter de s'ouvrir à l'épreuve d'une double interrogation, interne et externe à son champ disciplinaire, et pour ce faire il m'a paru intéressant de proposer cette distinction entre une pluridisciplinarité qui accumule mais n'articule pas, une transdisciplinarité qui traverse les lieux de pertinence mais peut en faire perdre leur horizon, et une interdisciplinarité qui échange, coopère, partage en interrogeant et intégrant de façon critique. » (9, page 220)

La pluridisciplinarité est donc un état de fait, l'entremise de différentes compétences sans l'innovation intrinsèque au croisement, à la fusion compromettante des idées et des angles d'approche. Il n'y a, dans la pluridisciplinarité, aucune prise de risque. Plutôt qu'un idéal, elle peut être une étape menant à l'objectif de transdisciplinarité comme processus fécond. La croisée des disciplines peut à la fois être un espace de conflit et d'affirmation de dogmes, donc stérile, et à la fois être un espace de remise en question, d'identification de biais et un lieu d'émergence d'approches nouvelles à l'abri des guerres de chapelle. RESWEBER offre une différente perspective sur la pluridisciplinarité, en mettant l'accent sur les frictions et oppositions inhérentes au croisement des disciplines « [La pluridisciplinarité] est la mise à l'épreuve des pouvoirs inhérents à chaque discipline qui a tendance, de par sa clôture, à ériger des conclusions provisoires en dogmes ou en vérités. » (63, page 176) Il accentue cette vision conflictuelle avec son approche de l'interdisciplinarité, qui selon lui « se définit par un échange de concepts entre diverses disciplines, mais [...] se réalise, plus profondément, par le choc ou l'opposition des méthodes qu'elle provoque. » (63) Le [Tableau 1.1](#) détaille la distinction entre ces trois concepts.

TABLEAU 1.1 – Les niveaux d'intégration à la croisée des disciplines

Niveau	Concept	Définition
0	Hyperdisciplinarité	Concept proposé par MORIN pour décrire la tendance des disciplines à l'isolement et à un certain esprit de concurrence et de territorialité. Dans cette posture, l'expert refuse toute ingérence dans son domaine qu'il va juger intrusive. C'est un esprit défensif qui est le principal frein à la croisée des disciplines [53].
1	Pluri ou multidisciplinarité	Sur un objet d'étude particulier, juxtaposition de points de vue issus de différentes disciplines. C'est le niveau minimal car il n'y a pas, à ce stade, de compréhension réciproque du jargon relatif à chaque expertise ni d'objectif clairement établi de le partager. Les disciplines sont « alignées de façon parallèle », en silo, dans un mode « séquentiel » [45].
2	Interdisciplinarité	Partage d'expertises sur un objet d'étude, avec interpénétration par vulgarisation et pédagogie. Dans ce processus, les experts enrichissent leurs connaissances et font l'effort de séparer les définitions de concepts nomades relativement à chaque discipline. KLEIN distingue l'interdisciplinarité méthodologique — dont l'objectif est d'améliorer la qualité des résultats — de l'interdisciplinarité théorique beaucoup plus épistémologique, visant à travailler sur des niveaux d'abstraction holistiques et des cadres conceptuels. Elle note aussi que « interdisciplinarité » est souvent galvaudé pour désigner en fait une situation de pluridisciplinarité. L'interdisciplinarité étroite rapproche deux disciplines proches, comme la littérature et l'histoire, tandis que l'interdisciplinarité étendue rapproche des disciplines très différentes, comme sciences pures et sciences humaines [45].
3	Transdisciplinarité	Croisée et dépassement — transcendance — de disciplines qui peut amener à une reformulation critique des paradigmes acceptés dans chacune d'entre elle. En cela c'est un processus « violent » car elle exige, observe RESWEBER une grande tolérance à la remise en question [63] — on parle de transdisciplinarité « transgressive ». La démarche est épistémologique et, parfois, l'horizon de travail peut être perdu à force de manipuler les concepts [9]. KLEIN décrit dans la transdisciplinarité des niveaux d'intégration et de coopération maximaux [45]. Cette interpénétration élevée conduit à l'émergence de « nouvelles catégories conceptuelles et [à] une unification méthodologique » (45).

1.2 Solutions d'assistance à l'analyse comportementale par codage vidéo

Les assistants à l'analyse comportementale permettent l'observation et l'analyse du comportement d'animaux — humains inclus — à partir de diverses sources d'acquisition. L'usage typique consiste à l'enregistrement puis l'analyse d'activités, de postures, de mouvements, de positions et d'interactions sociales. L'éthologie, la psychologie, les sciences cognitives et l'ergothérapie sont des disciplines particulièrement adaptées à l'utilisation de tels outils. Un scénario d'utilisation simple pourrait être décrit ainsi : un analyste observe un ou plusieurs individus à un endroit particulier, et saisit des observations comportementales dont les valeurs possibles ont été définies dans une phase antérieure.

1.2.1 Description générale

Deux logiciels d'analyse comportementale se distinguent sur ce marché. Ils sont très spécialisés, et leurs licences sont onéreuses, chiffrées à plusieurs dizaines de milliers de dollars canadiens. *Observer XT* [55], produit par Noldus®, est le plus complet et le plus populaire. La compagnie s'est spécialisée dans les solutions d'analyse comportementale, dans un premier temps animale puis humaine. *Captiv* [70] offre des fonctionnalités très similaires, dont la comparaison avec le précédent sera faite plus tard. On peut aussi noter la présence de deux projets *Open Source* d'origine académique, aux fonctionnalités relativement limitées comparées à ces deux concurrents. Il s'agit de *Boris* [24] et *Anvil* [44].

Les deux solutions payantes offrent un cadre général de travail très similaire¹. Un *projet* regroupe un ensemble d'*observations*. Pour chacune d'entre elles, des données peuvent être associées. Ces données peuvent venir de différents types de capteurs, en général propriétaires. *Observer XT* offre la possibilité d'interfacer des systèmes d'acquisition de données physiologiques avec des dispositifs comme MindWare ou BIOPAC, et ouvre la voie à des technologies non-propriétaires. C'est alors l'horloge du système hôte (l'ordinateur) qui est utilisée pour garantir la synchronicité. Le scénario de travail d'un utilisateur est divisé en trois parties : (1) *Configuration*, (2) *Observation* et (3) *Analyse*. Dans la première phase, l'utilisateur configure notamment un *schéma de codage*², qui va structurer les données d'observation et contraindre leur saisie, et éventuellement des *variables indépendantes*, qui permettent d'associer une va-

1. Dans cette section, le vocabulaire du domaine applicatif propre à ces solutions sera donné en *italique*, traduit de l'anglais assez littéralement.

2. Dans ce contexte, *codage* désigne l'action de fournir des données observationnelles.

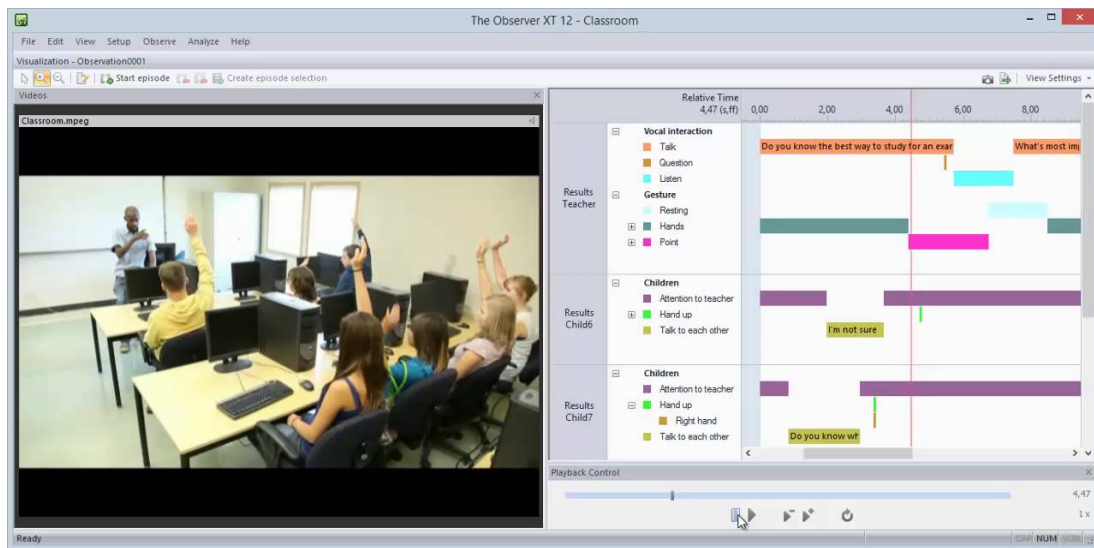


FIGURE 1.1 – Capture d'écran d'Observer XT 12

riable à chaque *observation*, ainsi que des *paramètres d'observation*. Dans la deuxième phase, les données sont récoltées et l'utilisateur fait la saisie d'*observations* à partir du *schéma de codage*. Dans la troisième phase, l'évaluateur peut utiliser des fonctionnalités de visualisation statistiques et d'exportation à partir de *sélecteurs de données*, qui sont des blocs logiques de sélection et de filtrage des données. Il peut aussi consulter les vidéos d'*observation* en lecture seule, avec bout-à-bout les observations de la phase précédente (Figure 1.1).

1.2.2 Analyse fonctionnelle détaillée

Schéma de codage

Le *schéma de codage* contraint la structure du *codage* qui va être effectué dans la phase d'*observation*. Des éléments de *modèle analogique* sont proposés, comme la définition d'ensemble de *sujets* de l'observation, des *comportements* de ces sujets et de propriétés relatives à ces *comportements*, appelés *modificateurs*. Les *sujets* sont des individus avec un nom générique. Lorsqu'un seul *sujet* est observé, il n'est pas nécessaire de le définir. Dans le cas où plusieurs *sujets* sont définis, les *comportements* doivent être attribués à un *sujet* dans la phase d'observation. Les *comportements* peuvent être définis de différentes manières. Soit en tant que *comportement à état*, caractérisé par un *temps de début* et un *temps de fin*; soit par un *comportement ponctuel*, localisé par un temps relatif par rapport au début de l'expérimentation. Ils doivent être définis dans un groupe, lequel peut être *mutuellement exclusif* — c'est-à-dire que deux comportements

de ce groupe ne peuvent être concurrents dans une même *observation* —, ou bien *non mutuellement exclusif*. Des touches peuvent-être affectées à chacun de ces comportement. Dans le cas d'un *comportement à état* issu d'un groupe *non mutuellement exclusif*, un appui répété sur une touche va successivement marquer le début puis l'arrêt d'un comportement. Dans le cas d'un *comportement à état* issu d'un groupe *mutuellement exclusif*, un appui va marquer le basculement d'un état à un autre. Dans le cas d'un *comportement ponctuel*, un appui va créer une instance de ce comportement. Il est aussi possible de restreindre des *comportements* à certains sujets.

Les *modificateurs* peuvent être définis pour un *sujet* ou un *comportement*. Ils sont soit *nominiaux* — c'est-à-dire qu'ils peuvent prendre un ensemble fini de valeurs labellisées —, soit *numériques*.

Configuration de projet

Dans *Observer XT*, le *projet* peut être configuré de différentes manières. Le choix d'une *source d'observation* se fait en amont, et permet soit un *codage* temps-réel (*observation en direct*), soit un *codage a posteriori* (*observation hors-ligne*). Seul le mode d'observation temps-réel permet de configurer l'acquisition de données de capteurs. Mais dans ce cas les seules sources vidéos disponibles sont des caméras directement connectées à l'ordinateur par USB ou Firewire³. Le mode d'observation peut être modifié pour une observation en particulier. Des *variables d'utilisateur* peuvent être définies, identifiées par un *label* et décrites par un *type* (numérique, logique, durée, estampille temporelle, texte). Pour le type numérique, l'unité peut être donnée. Leur *portée* désigne « la portion de l'observation sur laquelle s'applique la variable ». Elle peut être l'*observation* elle-même ou encore un parmi les *sujets* définis dans le *schéma de codage*.

Statistiques

Les fonctionnalités statistiques sont accessibles à partir de *sélections de données*. Un outil de sélection de données est donc proposé pour ensuite générer des rapports statistiques. Les fonctions définies sont le *groupement de données*, qui permet de grouper des observations (*expérimentations* dans COLAB), le *filtrage de données* qui permet de sélectionner, pour une expérimentation, les propriétés à récolter et enfin les *intervalles de temps* qui permettent de limiter la

3. Noldus offre une solution propriétaire, nommée Viso [56], qui permet l'enregistrement multi-source dans un environnement particulier.

sélection à une portion de chaque expérimentation. Ces sélecteurs peuvent être combinés pour former des sélecteurs complexes.

1.2.3 Comparaison

Le **Tableau 1.2** répertorie les fonctionnalités et caractéristiques-clé de ces systèmes concurrents. Dans ce tableau, les fonctionnalités qui sont supportées mais limitées par une technologie propriétaire sont marquée de la lettre « P ». Les fonctionnalités que nous avons identifiées dans cette comparaison ont pour but d'offrir des services d'acquisition, d'analyse et de collaboration interdisciplinaire. Il est notable que les systèmes existants axent leurs fonctionnalités sur l'acquisition et l'analyse sans offrir aucune fonctionnalité de collaboration.

TABLEAU 1.2 – Matrice de concurrence fonctionnelle

Fonctionnalité ou caractéristique	O.XT 12	Captiv
Dispositifs d'acquisition		
Capteurs physiologiques	✓	P
Capteurs environnementaux	✗	P
Capteurs biomécaniques	✗	P
Événements systèmes d'interactions homme-machine	✓	✓
Caméra USB ou Firewire	✓	✓
Caméras sur IP par RTSP	P	✗
Importation différée de données		
Importation de fichiers vidéo	✓	✓
Interfaçage avec environnement intelligent	✗	✗
Vidéos		
Support de vidéos panoramiques	✗	✓
Sources vidéos multiples	2	✓
Vignettes vidéos lors d'un survol de la barre d'avancement	✗	✗
Acquisition vidéo temps-réel	✓	✓

— suite à la page suivante —

Matrice de concurrence fonctionnelle – suite

Fonctionnalité ou caractéristique	O.XT 12	Captiv
Disponible en ligne	✗	✗
Collaboration entre chercheurs	✗	✗
Visualisation de graphes d'activité synchrones avec la vidéo	✓	✓
Analyse		
Transcript, verbatim	✗	✗
Définition de comportements	✓	✓
Comportements mutuellement exclusifs en option	✓	?
Catégorisation des comportements	✓	?
Séparation entre annotations descriptives et interprétatives	✗	✗
Définition d'une grille d'évaluation	✗	✗
Définition d'un motif d'observation	✓	✓
<i>Mapping</i> de comportements à des touches de clavier pour le codage	✓	✓
<i>Mapping</i> de sujets à des touches de clavier pour le codage	✓	✓
Visualisation de statistiques	✓	✓
Rapports		
Éditeur de rapports	?	✓
Export de rapports au format CSV	?	✓
Export de rapports au format EXCEL	?	✓
Collaboration		
Partage d'outils d'analyse entre chercheurs	✗	✗
Facilitation de l'analyse en aveugle (floutage d'expérimentation)	✗	✗
Transparence des évaluations et identification de l'auteur d'une annotation	✗	✗
Support des évaluations inter-juges et réflexives	✗	✗
Référence facilitée à un moment d'expérimentation	✗	✗

TABLEAU 1.3 – Objectifs du Profil des AVQ

Mise en situation
<ol style="list-style-type: none"> 1. Évaluer la capacité de réalisation des Activités de la Vie Quotidienne (AVQ). 2. Évaluer la nature et l'importance des difficultés reliées aux AVQ.
Questionnaire
<ol style="list-style-type: none"> 3. Évaluer les perceptions de la personne et d'une personne significative concernant : <ul style="list-style-type: none"> — la capacité de réalisation des AVQ; — la capacité de prise de responsabilités; — la capacité de planification, d'organisation et d'initiative; — les changements (degré de participation et prise de responsabilités); — les difficultés (nature, importance et conséquence pour les AVQ). 4. Évaluer le degré de satisfaction de la personne et d'une personne significative face au rendement dans les AVQ. 5. Évaluer l'impact des difficultés sur les habitudes de vie de la personne et d'une personne significative.

Source : DUTIL, É. et al., 2005, page 10 [16]

1.3 Le profil des AVQ, une méthodologie clinique d'évaluation de l'autonomie

1.3.1 Introduction

Il existe à ce jour deux manières d'évaluer l'autonomie liée aux habitudes de vie, la première par des questionnaires, la seconde par une mise en situation dans laquelle la personne réalise des **AIVQ**. Si la première est rapide et peu coûteuse, la seconde est considérée dans la littérature comme plus fiable et plus fine, même si elle demande de déployer une logistique beaucoup plus conséquente.

Une des principales raisons qui expliquent cette différence est l'attitude souvent biaisée des personnes ou de leurs aidants qui les pousse à surévaluer leur autonomie et à minimiser les

difficultés qu'ils rencontrent au quotidien lorsqu'ils répondent au questionnaire.

Le profil des **AVQ** est une approche mixte qui a été développée pour les personnes ayant eu un traumatisme crânien. Il comprend « une mise en situation et un questionnaire administré sous forme d'entrevue semi-structurée. » (16, page 9) Ces deux modes d'évaluation juxtaposés permettent la comparaison entre l'autonomie perçue par le participant, recueillie dans un entretien semi-structuré, et l'autonomie réelle, en situation, évaluée par des observateurs spécialisés (**ergothérapeutes**). L'objectif clinique est d' « évaluer l'indépendance de la personne dans la réalisation de ses activités de la vie quotidienne. »

1.3.2 Mise en situation

Dans la mise en situation, les principales aptitudes en jeu pour qualifier l'autonomie sont la capacité de décider, de planifier, de réaliser des tâches et éventuellement de se corriger (**Tableau 1.4**). Ces aptitudes sont évaluées pour chaque tâche que le participant effectue à l'exception de l'opération « formuler un but » qui n'est pas pertinente lorsque l'examineur demande explicitement à la personne de réaliser une tâche précise. Une tâche est définie comme une activité qui nécessite une certaine planification pour l'exécution, et qui peut être décomposée en une séquence de sous-tâches ou actions. La granularité d'une telle tâche est variable et les auteurs ne fournissent pas d'ordre de grandeur en particulier.

Préparation

Pour « accroître la validité écologique », les auteurs recommandent d'effectuer la mise en situation à domicile, et seulement en dernier recours dans un établissement. Vingt tâches⁴ sont répertoriées [17, page 5]. Pour chacune d'entre elles, une liste de sous-tâches ainsi qu'une liste du matériel nécessaire à sa mise en œuvre sont fournies (un exemple complet est disponible en **Annexe D.2**). L'examineur-accompagnateur encadre l'activité. Il doit « fournir le moins de structure et de consignes possibles à la personne, de façon à obtenir le maximum d'informations sur ses capacités » (16, page 23). Les auteurs laissent à la discrétion des cliniciens les conditions limites d'intervention, et dans un cadre de recherche, ces conditions devraient être

4. Les tâches détaillées sont les suivantes : (1) faire sa toilette ; (2) prendre soin de son apparence ; (3) s'acquitter de l'hygiène excrétrice ; (4) mettre ses vêtements et ses chaussures ; (5) prendre un repas ; (6) respecter sa diète/prendre sa médication ; (7) préparer un repas léger ; (8) préparer un repas chaud ; (9) faire le ménage quotidien ; (10) faire le ménage hebdomadaire ; (11) entretenir les vêtements ; (12) se déplacer à l'extérieur ; (13) utiliser le transport en commun régulier ; (14) conduire une automobile ; (15) faire des courses ; (16) téléphoner pour une information ; (17) s'acquitter d'une facture ; (18) utiliser un guichet automatique ; (19) faire un budget ; (20) respecter ses rendez-vous.

explicitement communiquées aux examinateurs pour atténuer la variabilité interjuge. L'examineur doit avoir en sa possession :

- la « fiche d'identification », qui contient les informations générales sur la personne, son milieu de vie et ses modes de déplacement ;
- pour chaque tâche prévue (un exemple complet est disponible en [Annexe D.2](#)) :
 - la « fiche descriptive de tâche » ;
 - la « fiche d'observation des opérations », qui vont servir de base pour la rédaction du rapport d'évaluation écrit ;
 - le « formulaire de cotation sommaire » ;
 - si applicable pour la tâche, la feuille de correction associée (tâches 16, 17 et 19).

La [cotation de grille](#) est une étape importante qui consiste à attribuer une appréciation générale sur l'autonomie de la personne d'une manière relativement reproductible. Pour permettre cette reproductibilité, des critères clairement établis sont définis dans l'évaluation de chaque tâche. L'échelle utilisée est de type ordinal à quatre niveaux de 0 à 3, les valeurs supérieures correspondant à une situation de non-applicabilité [Annexe D.1](#). Cette échelle est appliquée pour chacune des opérations de tâche ainsi que pour la tâche en général. Il y a donc quatre à cinq cotes par tâche suivant que l'énonciation (« formuler un but ») est applicable ou non. L'évaluateur doit prendre en compte le contexte environnant pour identifier des « erreurs » de comportement, comme par exemple la température extérieure dans la tâche n°4 « Mettre ses vêtements et chaussures ». Il doit aussi tenir compte d'autres éléments de contexte importants comme l'identité socioculturelle de la personne, ses habitudes de vie d'avant-traumatisme, et enfin « le degré de nouveauté et de familiarité en regard de la tâche et l'environnement où s'effectue l'expérimentation ». Il n'existe pour le moment pas de cote cumulative dans le profil des AVQ, c'est-à-dire une cote qui permet d'évaluer l'autonomie globale de la personne à partir des tâches étudiées. C'est donc à l'appréciation des cliniciens de faire une évaluation globale quantitative et qualitative, qui sera détaillée dans les chapitres suivants sous le nom de [grille d'évaluation](#).

1.3.3 Entrevue

Pour l'entrevue, l'emphasis est mise sur les points suivants :

- la perception de l'autonomie par le participant et l'aidant ;
- les responsabilités assumées par le participant et par l'aidant ;
- les initiatives prises par le participant.

Le questionnaire comprend quatorze questions, et il a pour but d'évaluer, à partir du point de vue du patient et d'un proche ou aidant naturel⁵, les évolutions entre la situation *pré* et post-traumatique. Il est notamment demandé aux personnes interrogées si elles perçoivent un changement de la fréquence de la pratique d'activités, dans la capacité d'assumer différentes responsabilités ou de planifier des activités.

1.3.4 Applicabilité au dépistage de la maladie d'Alzheimer

Le profil des **AVQ** est à l'origine destiné à évaluer l'autonomie de personnes souffrant de traumatismes crâniens. Les principaux symptômes identifiés sont des troubles de l'équilibre, de la coordination, des problèmes de dextérité et de mémoire, des troubles de l'attention, et d'une manière générale des **fonctions exécutives**, c'est-à-dire « la capacité à planifier et réaliser des comportements complexes orientés vers des buts ». Si les auteurs recommandent une observation à domicile, son usage est détourné dans le projet **DEI** puisqu'il devient un outil pour détecter les marqueurs précoces de la **Maladie d'Alzheimer (MA)**. En effet, plusieurs recherches semblent indiquer que des marqueurs très précoces de la **MA** peuvent apparaître jusqu'à dix ans avant le déclenchement des premiers symptômes. Ces indices se retrouvent dans les fonctions d'organisation et de planification, comme la gestion d'un budget, les errances, la survenance de difficultés à utiliser certaines technologies maîtrisées, la prise de médicaments ou encore le déplacement en ville. La **MA** semble affecter dans un premier temps les capacités d'adaptation dans des situations non familières [59]. C'est pourquoi le protocole de recherche du projet **DEI** privilégie des expérimentations en laboratoire, le participant n'ayant pas une connaissance préalable de l'environnement où il sera évalué.

5. Les auteurs parlent de « personne significative ».

TABLEAU 1.4 – Définition des opérations dans le Profil des AVQ

Opération	Définition
1. Formuler un but	Capacité de : <ul style="list-style-type: none"> — trouver une solution pour satisfaire un besoin ou résoudre une situation problématique.
2. Planifier	Capacité de : <ul style="list-style-type: none"> — réfléchir avant d'agir aux conditions de départ ; — identifier les alternatives ; — choisir l'alternative la plus adéquate parmi plusieurs ; — élaborer un plan d'action stratégique général et un plan tactique (séquence d'actions et d'étapes).
3. Exécuter	Capacité de : <ul style="list-style-type: none"> — initier son plan d'action ; — réaliser le plan d'action tout en réagissant aux erreurs et aux situations nouvelles ; — percevoir les erreurs de planification et d'exécution ; — modifier l'exécution de la tâche en fonction des erreurs et des changements survenus.
4. S'assurer de l'atteinte du but initial	Capacité de : <ul style="list-style-type: none"> — vérifier que la tâche prévue initialement a été accomplie ; — comparer le résultat final au but initial ; — accepter ou rejeter le résultats ; — mettre fin au processus ou recommencer si le résultat est rejeté.

Source : DUTIL, É. et al., 2005, page 16 [16]

1.4 Systèmes de contrôle d'accès

Dans les technologies de l'information, les systèmes de contrôle d'accès permettent d'identifier les prérogatives associées à chaque type d'utilisateur afin de s'assurer que chacun d'entre eux n'ait accès qu'à un ensemble pertinent d'informations et d'actions dans un contexte donné. En sécurité informatique, le principe sous-jacent est le principe du moindre privilège (*principle of least privilege*), qui énonce que pour un utilisateur donné, une analyse de son activité doit permettre de déterminer l'ensemble minimal de privilèges qui doit lui être accordé : « By denying to subjects transactions that are not necessary for the performance of their duties, those denied privileges cannot be used to circumvent the organizational security policy. » ⁶ (21, page 562) Comme nous le verrons plus tard, certaines restrictions sont obligatoires pour satisfaire des contraintes liées au **domaine d'application**, par exemple l'évaluation à l'aveugle. Dans le cadre de l'application proposée dans cette maîtrise, le principe du moindre privilège ne doit pas être compris comme étant un principe motivé par une défiance vis-à-vis des différents utilisateurs potentiels de l'application, mais plutôt comme une garantie qu'une erreur humaine ait un impact aussi faible que possible sur l'**intégrité** des travaux réalisés par son entremise. Il permet aussi de garantir la confidentialité de certaines données, pour se conformer aux politiques d'éthique de la recherche.

De nombreux états de l'art dans le domaine des modèles de contrôle d'accès ont déjà été proposés (R. CROOK, D. INCE et B. NUSEIBEH [13], L. FUCHS, G. PERNUL et R. SANDHU [26] et C. FELTUS [20]). La littérature fait la distinction entre différents types de systèmes de contrôle d'accès, dont un inventaire sera dressé.

1.4.1 Typologie des systèmes d'accès

La gestion de l'accès aux ressources est un besoin crucial pour beaucoup d'entreprises et organisations.

Ce besoin a été historiquement formalisé dans les années 1970-1980 par le **Department of Defense (DoD)** étasunien via le standard **Trusted Computer System Evaluation Criteria (TSEC)**. Dans l'armée, la ressource est avant tout de nature informationnelle plutôt que fonctionnelle. Les premiers systèmes d'accès sécurisés ont donc été calqués sur la politique de sécurité propre à l'institution militaire. Le **TSEC** explicite le couplage fort entre la politique de confidentialité de l'armée et celle du système d'information qui doit la refléter [14].

6. En interdisant aux sujets des transactions qui ne sont pas nécessaires à l'exercice de leurs fonctions, les privilèges ainsi déclinés ne peuvent être utilisés pour contourner la politique de sécurité de l'organisme. — traduction libre

Ce document de référence formalise la distinction entre différentes modalités de contrôle d'accès, et notamment entre **Contrôle d'Accès Obligatoire (CAO)** et **Contrôle d'Accès Discretionnaire (CAD)**. Avant de résumer les principales différences entre ces deux modalités, il est nécessaire d'apporter quelques définitions relatives au contrôle d'accès. Dans un tel contexte, un utilisateur souhaite accéder à un objet, c'est-à-dire une ressource. L'utilisateur se sert d'un programme par lequel il s'identifie. Le programme qui exécute les instructions de l'utilisateur est appelé sujet⁷. L'objet auquel il souhaite accéder est *a priori* détenu par l'organisme qui exerce le contrôle d'accès, bien que la possession (*ownership*) de cet objet puisse être déléguée à un utilisateur privilégié dans certaines politiques d'accès. Le système va garantir ou refuser l'accès suivant sa politique d'accès. Le sujet doit avoir une permission (ou suivant les auteurs, « autorisation », « privilège ») portant à la fois sur l'objet et sur la modalité d'accès (suivant les auteurs, est désigné comme « action », « transaction », « opération » ou encore « méthode ») pour que sa demande d'accès soit garantie.

Dans le cas de l'application d'une politique **CAO**, une seule autorité délivre des droits d'accès et les utilisateurs ne sont pas autorisés à les modifier. Une telle politique est définie comme

« a mean of restricting access to objects based on the sensitivity (as represented by a label) of the information contained in the objects and the formal authorization (i.e., clearance) of subjects to access information of such sensitivity. »⁸ (14, page 109)

Le **Lattice Based Access Control (LBAC)** est une implémentation de **CAO** dans laquelle les permissions sont accordées suivant une règle simple : le niveau de confidentialité de la ressource accédée doit être égal ou inférieur au niveau de confidentialité maximal conféré à l'utilisateur. Ce niveau de confidentialité est appelé niveau de sécurité (*security level*), par exemple *TopSecret*, *Secret*, *Confidential* et *Unclassified*. L'usage du **LBAC** est particulièrement courant dans les institutions militaires.

Dans une politique **CAD** sont définis les concepts de sujet, d'objet et d'action. Cette dernière correspond à la modalité d'accès à l'objet, par exemple lecture, écriture. Dans un tel contexte, l'action détermine l'ensemble des permissions qu'un sujet peut avoir sur un objet. Contrairement au **CAO**, l'autorité centrale n'a pas le monopole du contrôle d'accès, qui peut être délégué en

7. Même si dans la littérature, « sujet » et « utilisateur » sont souvent confondus, la distinction est importante, car un sujet peut tout à fait agir au nom d'un utilisateur usurpé, par exemple si les identifiants (*credentials*) de l'utilisateur ont été subtilisés. Autre nuance : « sujet » englobe une définition plus large puisqu'il désigne n'importe quel acteur d'un système, comme un agent, un script ou un serveur tiers.

8. *un moyen de restreindre l'accès aux objets en fonction d'une sensibilité – définie par une étiquette – des informations contenues dans les objets et de l'autorisation formelle confiée aux sujets pour accéder à des informations de cette même sensibilité.* — traduction libre

affectant à un sujet des droits de contrôle d'accès (*control access rights*). La caractéristique fondamentale est donc que les permissions affectées à un sujet peuvent porter sur l'allocation de permissions à d'autres sujets. C'est le sens de *discretionary*, c'est-à-dire discrétionnaire, « à la discrétion de ».

1.4.2 Role Based Access Control

RBAC nowadays marks the de facto standard in enterprise systems

— L. FUCHS, G. PERNUL et R. SANDHU [26, page 2]

Définition originale

Présenté en 1992 lors d'une conférence sur la sécurité des systèmes d'information par D. F. FERRAIOLO et D. R. KUHN [21], le **Contrôle d'Accès à Base de Rôle (CABR)** est une évolution des systèmes de contrôle d'accès vers une plus grande flexibilité et une meilleure mise à l'échelle, particulièrement adaptée aux organismes qui ne sont pas issus du domaine militaire. En effet, ce nouveau modèle précise la correspondance entre la structure hiérarchique de l'organisme et la répartition des permissions. Cette correspondance se fait au travers des rôles, qui sont définis comme un ensemble de permissions rattachées à une fonction hiérarchique plutôt qu'à un individu. Ce dernier, par son appartenance à un rôle, se verra donc accorder des accès : « A role can be thought of as a set of transactions that a user or set of users can perform within the context of an organization. »⁹ (21, page 557)

Ici, la notion de transaction se réfère à une procédure de transformation, d'entreposage ou de lecture de données [21, p.557]. Elle peut donc désigner un ensemble d'effets dont l'articulation reflète la sémantique du **domaine d'application**. D'ailleurs, le modèle **CABR** peut être résumé par un syntagme sujet-verbe-objet, le sujet étant l'utilisateur, le verbe une transaction liée à un rôle et l'objet la ressource. C'est pourquoi *subject* et *user* sont utilisés de façon interchangeable par FERRAIOLO et KUHN⁷.

Sur la question de savoir si **CABR** est plutôt **CAD** ou **CAO**, SANDHU et al. considèrent qu'il est agnostique : « Recall that by itself, RBAC is policy neutral ; however, individual RBAC

9. Un rôle peut être considéré comme un ensemble de transactions qu'un utilisateur ou un ensemble d'utilisateurs peut effectuer dans le contexte d'une organisation. — traduction libre

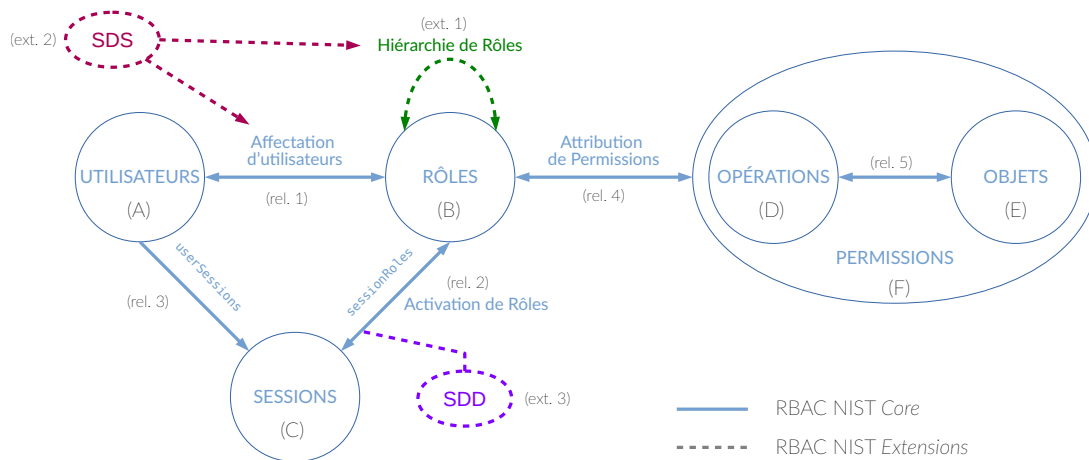


FIGURE 1.2 – Modèle RBAC NIST 2001 et ses extensions

Inspiré de : FERRAILOLO, D. F., KUHN, D. R. et CHANDRAMOULI, R., 2001, pages 232–239 [22]

configurations can support a mandatory policy, while others can support a discretionary policy. »¹⁰ (65, page 40)

Neuf ans plus tard et après de nombreuses publications dans le domaine, une proposition standardisée du modèle **CABR** comprenant quatre variantes est proposée par D. F. FERRAILOLO, D. R. KUHN et R. CHANDRAMOULI [22].

Le modèle qui factorise les caractéristiques des trois autres est appelé *Core RBAC* (Figure 1.2). Ce modèle est décrit par un ensemble minimal de caractéristiques nécessaires à sa conformité.

Les auteurs définissent des propriétés associées à leur modèle minimal : [P1] Pour chaque utilisateur (Figure 1.2, item A), l'ensemble des rôles actifs (*rel. 2*) correspond à ceux qu'un utilisateur a activé dans une session (item C). [P2] Chaque utilisateur peut être membre d'un ou plusieurs rôles (item B). L'ensemble de rôles affectés à un utilisateur (*rel. 1*) est appelé « rôles autorisés ». [P3] Un rôle peut conférer le droit d'effectuer une ou plusieurs opérations (item D) sur un ou plusieurs objets (item E). Ces droits qui lient une opération à un objet sont appelés permissions (item F).

Trois règles sont dérivées de ces propriétés : [R1]. Affectation d'utilisateurs (*rel. 1*) : un utilisateur peut exécuter une opération seulement s'il s'est vu assigné un rôle. [R2]. Activation de rôles (*rel. 2*) : les rôles actifs d'une session doivent constituer un sous-ensemble des rôles autorisés.¹¹ [R3]. Attribution de permissions (*rel. 4*) : un utilisateur peut exécuter une opération

10. Rappelons qu'en tant que tel, **CABR** est politiquement neutre ; cependant, ses configurations individuelles peuvent entériner une politique « obligatoire » (mandatory), tandis que d'autres peuvent mettre en oeuvre une politique « discrétionnaire ». — traduction libre

11. Il est à noter que dans les premières définitions, notamment par D. F. FERRAILOLO et D. R. KUHN [21],

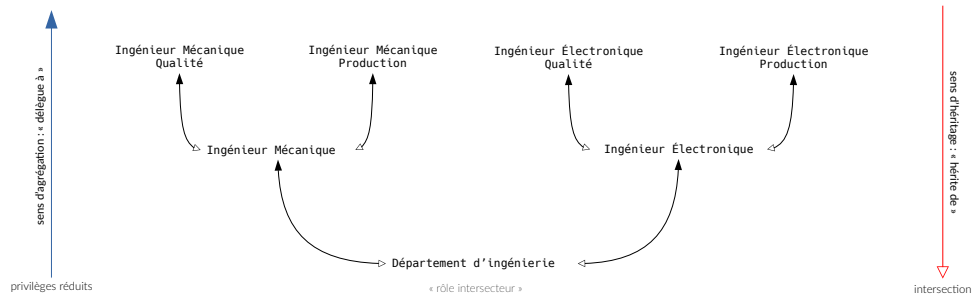


FIGURE 1.3 – Hiérarchie de rôles en structure d'arbre

Inspiré de : FERRAILOLO, D. F., KUHN, D. R. et CHANDRAMOULI, R., 2001, page 236 [22]

op_u sur un objet ob_v seulement si une permission $p(op_u \rightarrow ob_v)$ liant op_u à ob_v existe pour au moins un des rôles actifs de l'utilisateur. Il est à noter que le standard ici défini ne donne aucune contrainte sur la nature statique ou dynamique de la résolution des permissions, c'est-à-dire sur leur portée (*scope*, *rel.* 5).

Hiérarchie de rôles

L'extension « hiérarchie de rôles » (Figure 1.2, *ext. 1*) permet de définir des relations entre rôles.

Elle a pour intérêt de refléter la structure fonctionnelle d'un organisme, en autorisant la factorisation des permissions par héritage. Les auteurs distinguent trois structures d'héritage possibles :

- *Hiérarchie en structure d'arbre* (Figure 1.3). Dans une telle hiérarchie, l'héritage multiple est proscrit. Le sens de parcours privilégié est le « sens d'agrégation », qui converge depuis les niveaux inférieurs vers la racine de l'arbre, qu'on nomme « rôle unificateur »¹² car il agrège les permissions de tous les autres rôles, plus ceux qui lui sont exclusifs.
- *Hiérarchie en structure d'arbre inversé* (Figure 1.4). Dans une telle hiérarchie, l'héritage multiple est autorisé. Le sens de parcours privilégié est le « sens d'héritage », qui converge

une session établissait une correspondance entre un utilisateur et un unique rôle. Ceci avait pour but de garantir le principe du moindre privilège. Mais cette restriction a été abandonnée par la suite car elle contrevient à la flexibilité du modèle.

12. En théorie des ensembles, ceci correspond à l'union d'une famille d'ensembles. Ici, chaque ensemble est un rôle et ses éléments sont des permissions. Chaque membre de cette famille est donc un rôle parmi tous les rôles décrivant la hiérarchie.

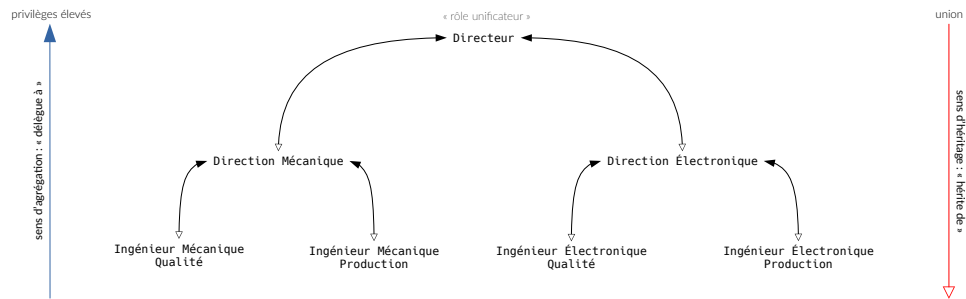


FIGURE 1.4 – Hiérarchie de rôles en structure d'arbre inversé

Inspiré de : FERRAILOLO, D. F., KUHN, D. R. et CHANDRAMOULI, R., 2001, page 236 [22]

depuis les niveaux inférieurs vers la racine de l'arbre, qu'on nomme « rôle intersecteur »¹³ car il factorise les permissions de tous les autres rôles.

- *Hiérarchie en structure de treillis* (Figure 1.5), c'est la combinaison des deux précédentes. L'héritage multiple y est autorisé.

La hiérarchie de rôle doit constituer une structure d'ensemble partiellement ordonné (*poset*) où l'héritage se fait dans un sens spécifié¹⁴, c'est à dire que les rôles les plus proches du haut

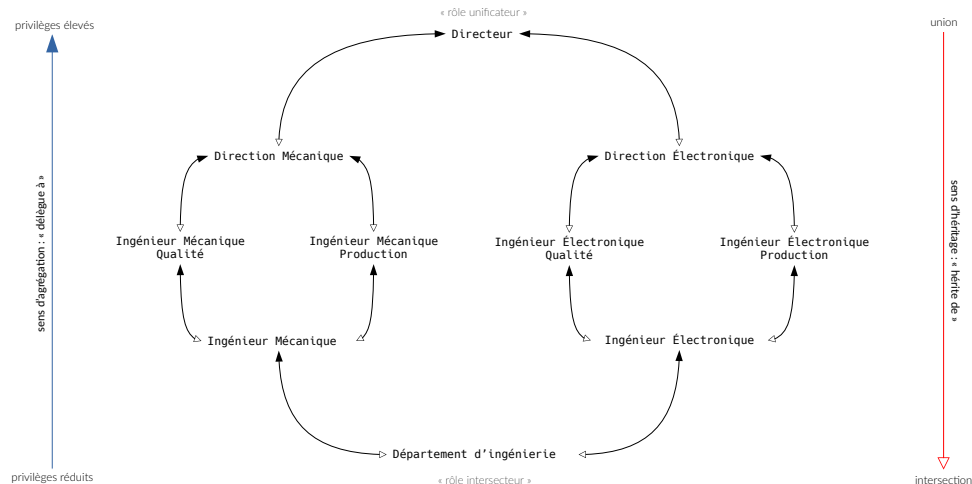


FIGURE 1.5 – Hiérarchie de rôles en structure de treillis

Inspiré de : FERRAILOLO, D. F., KUHN, D. R. et CHANDRAMOULI, R., 2001, page 236 [22]

13. En théorie des ensembles, ceci correspond à l'intersection d'une famille d'ensembles. Ici, chaque ensemble est un rôle et ses éléments sont des permissions. Chaque membre de cette famille est donc un rôle parmi tous les rôles décrivant la hiérarchie.

14. Le sens par lequel les permissions sont héritées semble *de facto* faire consensus, et l'usage veut que le sens d'héritage des permissions soit du haut vers le bas, c'est-à-dire que les rôles situés en haut héritent des privilèges des rôles situés plus bas. Mais ce sens devrait toujours figurer dans la légende du document pour lever toute ambiguïté. Les auteurs de ce modèle ne mentionnent pas ce problème, et l'initiative a donc été prise d'éclaircir cette zone d'ombre.

héritent des permissions des rôles situés plus bas.

À partir de ces observations, les auteurs définissent deux types de hiérarchie de rôles dans leur modèle **CABR** : une forme « générale » qui accepte l'héritage multiple, et une forme « limitée » qui n'autorise que l'héritage simple. Il est à noter qu'on peut déduire, par transitivité, des relations d'appartenance d'un utilisateur à un ensemble de rôles (*role membership*), et réciproquement d'un rôle à un ensemble d'utilisateurs (*user membership*) :

« the user membership set for a given role includes not only users directly assigned to that *given role* but also those users assigned to roles that inherit the given role. Analogously the role membership set for a given user includes not only roles directly assigned to the given user but also those roles inherited by the directly assigned roles. » ¹⁵ (22, page 244)

Ces propriétés stipulent donc que le sens d'héritage des utilisateurs appartenant à un rôle – ou sens d'héritage des membres – est opposé au sens d'appartenance des rôles à un utilisateur – ou sens d'héritage des permissions.

Contraintes supplémentaires

La séparation des devoirs (*separation of duties*) est une politique de sécurité qui stipule que pour une tâche critique donnée – c'est à dire un ensemble d'opérations qui aboutit à des effets qui peuvent mener à des conflits d'intérêts – un utilisateur du système ne doit jamais se voir assigné l'ensemble des permissions qui l'autoriseraient à compléter cette tâche. Pour réaliser cette politique, les auteurs proposent deux types de séparation des devoirs :

- à l'affectation d'utilisateurs (**Figure 1.2, rel. 1**), on parle alors de Séparation des Devoirs Statique (SDS) (**Figure 1.2, ext. 2**). Deux rôles peuvent être définis comme exclusifs, : dans ce cas de figure, un utilisateur ne peut se voir affecté deux rôles qui sont exclusifs entre eux.
- à l'activation de rôles (**Figure 1.2, rel. 2**), on parle alors de Séparation des Devoirs Dynamique (SDD) (**Figure 1.2, ext. 3**). Cette fois-ci, ce sont les rôles actifs de l'utilisateur dans une session qui peuvent être définis comme exclusifs.

15. *l'ensemble des utilisateurs appartenant à un rôle est défini non seulement par les utilisateurs directement affectés à ce rôle, mais aussi par les utilisateurs affectés à des rôles qui héritent dudit rôle. De manière symétrique, l'ensemble des rôles appartenant à un utilisateur est défini non seulement par les rôles directement assignés à cet utilisateur, mais aussi par ceux dont lesdits rôles héritent.* — traduction libre

1.5 Coopérativisme des plateformes

Le coopérativisme des plateformes est un mouvement civique né en 2014. Il promeut l'appropriation par la société civile de services en ligne, pour en faire des biens communs. L'initiative compte en juillet 2017 un peu moins de 200 plateformes.

1.5.1 Caractéristiques des plateformes coopératives

L'expression « plateformes coopératives » désigne à la fois le logiciel qui permet la mutualisation des moyens et la raison sociale qui représente les intérêts de ses bénéficiaires. Cette raison sociale n'est pas nécessairement de nature coopérative, et de nombreux modèles entrepreneuriaux existent. Mais, quelle que soit la structure adoptée, on peut retenir certains piliers qui sont autant de valeurs fondamentales. CHERRY reprend la liste des sept piliers des coopératives [11] :

- adhésion anti-discriminatoire et ouverte ;
- contrôle démocratique par les membres ;
- participation financière équitable parmi les membres ;
- autonomie et indépendance ;
- éducation, formation et information ;
- coopération inter-coopératives ;
- ambitions pour la communauté.

Ce dévouement à la communauté peut être transposé aux biens communs et épouser naturellement la mouvance décrite ultérieurement, dans la sous-section 1.5.4. En effet, la communauté, traditionnellement locale dans la coopérative classique, devient éventuellement nationale, voire internationale à travers la plateforme. Mais les objectifs restent identiques : servir l'intérêt collectif. L'idée que la possession, la gouvernance et les bénéfices soient partagés par les membres est prépondérante. Pour autant, une multitude de modèles sont envisageables. Le modèle mutualiste propose de partager la gouvernance ou la possession avec le client, *in fine* l'utilisateur. Le modèle coopératif fait ce partage avec les salariés.

1.5.2 Classification des modèles existants

La structure des modèles existants laisse apparaître une grande disparité. Après trois ans d'existence, le *Platform Cooperatives Consortium* a référencé 189 plateformes coopératives

dans le monde [61]. Parmi elles, 50 en Europe, 54 en Amérique du Nord, 7 en Australie et Nouvelle-Zélande et les quelques restantes disséminées sur le reste de la planète. On propose dans le **Tableau 1.5** un recensement des classes de plateformes avec une liste non exhaustive d'exemples. Elles prennent la forme de coopératives, de mutuelles ou d'organismes à but non lucratif exerçant des activités marchandes et sont exploitées conformément aux principes suivants :

- elles ont pour but de répondre aux besoins de leurs membres ou de la collectivité ;
- elles ne sont pas sous le contrôle décisionnel d'un ou de plusieurs organismes publics ;
- elles prévoient des règles de gouvernance démocratique par les membres ;
- elles aspirent à la viabilité économique ;
- elles prévoient des règles qui interdisent ou limitent la distribution des surplus générés par leurs activités ;
- en cas de dissolution, le reliquat de leurs biens est dévolu à un organisme ayant des objectifs semblables.

TABLEAU 1.5 – Classification des plateformes coopératives

Classe	Description	Exemples
Facilitateurs financiers	Ces organismes aident à la naissance et au soutien de coopératives ou de projets des communs. Elles facilitent l'émergence d'alternatives en donnant accès à des leviers de financement.	https://seed.coop https://gratipay.com https://snowdrift.coop https://fair.coop https://nycreic.com https://robinhoodcoop.org https://enspiral.co https://theworkingworld.org

— suite à la page suivante —

Classification des plateformes coopératives – *suite*

Classe	Description	Exemples
Désintermé- diateurs – DAOE (<i>Decentralised Autonomous Organiz-ation Enablers</i>)	Ces solutions utilisent des chaînes de blocs pour redéfinir les modalités contractuelles entre agents et parachever la désintermédiation. Elles permettent la création d'organisations auto-gérées (DAO) avec des règles de gouvernances qu'elles peuvent elles-mêmes (re)définir et entériner via la chaînes de blocs. Ces technologies sont disruptives car elles éliminent le besoin de médiation humaine dans l'élaboration de contrats.	http://ethereum.org/ http://backfeed.cc/ http://hexalina.io/ http://divvydao.org/ https://outlierventures.io/ https://www.bithouse.io/
Facilitateurs struc- turels	Ces organismes s'intéressent aux modalités légales de la raison sociale (statut de l'organisme) qui permet la mise en œuvre du coopérativisme des plateformes. Ils offrent un cadre légal qui rend la structure résiliente face à d'éventuelles tentatives de prédation (rachat par un investisseur tiers et détournement du « cap » de l'entreprise).	http://www.fairshares.coop/ http://purpose-economy.org
Plateformes de tra- vailleurs	Ces nombreuses structures réunissent les travailleurs indépendants d'une profession de service, comme les taxis, et mutualisent le coût de développement et de maintien d'un service en ligne et sur mobile.	http://www.alphataxis.fr/ http://co-optaxi.com/ http://www.cooptaxi.co.nz/ http://libretaxi.org/ https://loconomics.com/

— suite à la page suivante —

Classification des plateformes coopératives – suite

Classe	Description	Exemples
Plateformes pour artistes	De l'alternative à Spotify® rémunérant les artistes par chaînes de blocs à une banque de stockage de photos professionnelles, l'outil sert à la désintermédiation entre le consommateur et le producteur d'œuvres artistiques numériques, en assurant une juste rémunération des auteurs.	https://www.stocksy.com https://resonate.is https://tidal.com
Plateformes de collaboration	Ce sont des outils permettant la prise de décision ou la gouvernance d'un groupe.	https://loomio.com/

Ces chiffres témoignent d'un réel engouement de terrain pour proposer des alternatives au modèle économique classique. Deux tendances sont notables. D'une part, l'utilisation de chaînes de blocs comme outil naturel de désintermédiation. Une vingtaine de projets sont basés sur cette technologie et en tirent un potentiel disruptif important. D'autre part, une forte solidarité « coopérativiste ». En effet, trois classes parmi les six identifiées sont des facilitatrices (facilitateur financier, structurel et désintermédiaireur).

1.5.3 Les racines du mouvement

Code will present the greatest threat to both liberal and libertarian ideals, as well as their greatest promise. We can build, or architect, or code cyberspace to protect values that we believe are fundamental. Or we can build, or architect, or code cyberspace to allow those values to disappear. There is no middle ground. There is no choice that does not include some kind of building. Code is never found; it is only evermade, and only evermade by us.

— L. LESSIG [48, page 6]

Introduit par SCHOLZ en 2014, le mouvement du coopérativisme des plateformes (*platform cooperativism*) s'oppose aux intermédiaires de rente de l'économie du partage, comme Uber®, AirBnB®, Taskrabbit®, Amazon Mechanical Turk®, Foodora® ou encore Handy® [68]. La première inquiétude est de nature sociale. SCHOLZ reproche à ces compagnies de saboter le droit du travail des prestataires de service — par exemple les conducteurs de voiture

pour Uber, les livreurs de repas de Foodora et les jardiniers de Taskrabbit — en faisant d'eux des auto-entrepreneurs¹⁶ précarisés voire ultra-précarisés [57] ; dont l'exercice est contraint par les conditions d'utilisation de la plateforme, par laquelle ils peuvent accumuler des missions de travail. Cette précarisation ravive chez certains les témoignages des conditions ouvrières du XIX^e siècle où l'on payait à la journée [67], et la mise en concurrence des prestataires, prêts à transgresser la loi pour réaliser des missions [18]. La célèbre maxime *Code Is Law* de LESSIG s'applique parfaitement à cette situation, puisque ces compagnies, par l'intermédiaire de leurs plateformes, prennent le rôle États pour légiférer une partie du monde du travail. Cette critique est dans la continuité des travaux de SCHOLZ sur le *Digital Labour*¹⁷, proposé en 2012 [67].

La deuxième critique porte sur la « réintermédiation ». D'abord, l'émergence de ces plateformes a suscité beaucoup d'enthousiasme au cours des années 2000. D'une part parce qu'elles réalisaient l'idéal libéral d'agents économiques interagissant avec le moins de frictions possible : elles permettent des milliers de microtransactions, faisant de chaque utilisateur un micro-entrepreneur. D'autre part parce qu'elles nous rapprochent d'un idéal écologique où la mutualisation des moyens devient optimale, dans un environnement à ressources limitées. Le phénomène de rapprochement des individus, ou de suppression des intermédiaires dans un circuit de distribution a été désigné dès les années 1990 comme « désintermédiation ». Mais celle-ci est aujourd'hui contestée, comme le souligne KLEINER : « Disintermediating platforms were ultimately reintermediated by capitalist investors dictating that communications systems be built to capture profit. »¹⁸ (46) De nouveaux intermédiaires monopolistiques se sont ainsi nichés dans les espaces inoccupés de l'économie du partage. KLEINER fait l'analogie avec la colonisation illégitime de « nouvelles terres » initialement occupées par d'autres nations.

Le mouvement de concentration des acteurs privés du numérique décrit précédemment fait l'objet d'une critique plus large. Constatant les nouveaux pouvoirs conférés à ces entités non étatiques, de nombreux observateurs craignent un affaiblissement de l'État et de la société civile sous la pression de ces mastodontes qui assoient leur légitimité sur le « solutionnisme technologique », reliquat moderne de l'idéologie scientiste [54]. Cette conquête offensive du cyberspace par le marché — et dans une moindre mesure, par certains États — est une des principales motivations du mouvement du coopérativisme des plateformes. Ce dernier cherche à restituer une souveraineté spoliée à la société civile, en revendiquant son indépendance à l'égard du marché et

16. HILL décrit ce nivellement par le bas des salaires dans ce qu'il appelle la *freelance society* [37]. FRIEDMAN parle de *gig economy* [25].

17. Qui pourrait être traduit par « travail numérique », avec la connotation de pénibilité associée au *labor* en anglais.

18. *Les plateformes de désintermédiation ont finalement été « réintermédiées » par des investisseurs qui réclamaient que ces systèmes soient conçus pour drainer des bénéfices.* — traduction libre

de l'État [8]. Pour atteindre ce but, l'idée est de mener à leur terme les réflexions qui ont motivé la création de logiciels libres en s'ouvrant à la question de la gouvernance : concevoir, « implémenter » et collectiviser des plateformes de désintermédiation, dont la propriété et la gestion sont partagées entre bénéficiaires¹⁹ [11]. C'est donc, d'une certaine manière, la société civile qui se réapproprie modestement des parcelles de pouvoir dans le cyberspace, notre « Sixième Continent ».

Ces enjeux politiques autour du cyberspace font du génie logiciel une discipline résolument partisane, souvent à l'insu de ses disciples. C'est ce qu'affirme MOROZOV — dans la même ligne de pensée que LESSIG — lorsqu'il déclare dans un entretien que « les technologies sont des concentrés d'idéologies politiques » et qu'un pays peut perdre une part importante de sa souveraineté s'il ne maîtrise pas les outils dont il fait l'usage [2]. Nier cette réalité et le pouvoir de transformation extraordinaire des technologies sur nos êtres [1] reviendrait alors à s'abandonner au solutionnisme décrié précédemment ; une forme de bigoterie techno-béate. Assumer cette réalité et prendre position relève donc aussi bien de l'impératif moral que de l'esprit d'analyse scientifique. La technologie n'est et ne sera *jamais* neutre. L'emprise qu'elle peut susciter chez certains n'est pas le fruit du hasard, et il a été observé dans la Silicon Valley qu'il est courant d'exploiter les biais et les faiblesses psychologiques des individus pour maximiser leur temps d'utilisation [33].

Dans ce nouvel « Âge de la distraction », l'exploitation de l'attention humaine, devenue un nouvel objet de conquête marketing, est la cible de virulentes critiques. Des auteurs lient ces pratiques à la distractibilité croissante des plus jeunes générations [12, 27], ou encore au succès et à la profusion d'« infotainment », de fausse-information et de contenus racoleurs. L'attention devient elle aussi un enjeu pour nos démocraties et pour nos sociétés car, comme l'a souligné AIKEN, au contact de la technologie, notre « nature humaine » est étonnamment malléable [1]. L'intrusivité ambitionnée aujourd'hui par les acteurs privés est rapportée par BRIA lorsqu'elle affirme : « The builders of emerging online platforms aim to become pervasive across all productive sectors, and to permeate every level of society. »²⁰ (8) La récolte massive de données d'utilisateurs — le « profilage de clientèle » en termes marketing — et sa juteuse exploitation par les *data brokers*²¹ sont rapportées avec minutie par GOODMAN [28, chap. 5]. BOLLIER s'en inquiète et voit là un enjeu politique majeur pour le coopérativisme des plateformes : « We

19. Les « Bénéficiaires » sont d'une manière générale les utilisateurs de la plateforme, souvent membres de la structure. Suivant les situations ils peuvent être, par exemple, salariés, travailleurs ou indépendants.

20. *Les bâtisseurs de plateformes en ligne émergentes visent à s'immiscer dans l'ensemble des secteurs productifs et à imprégner tous les niveaux de la société.* — traduction libre

21. Qu'on peut traduire par « courtiers de données ».

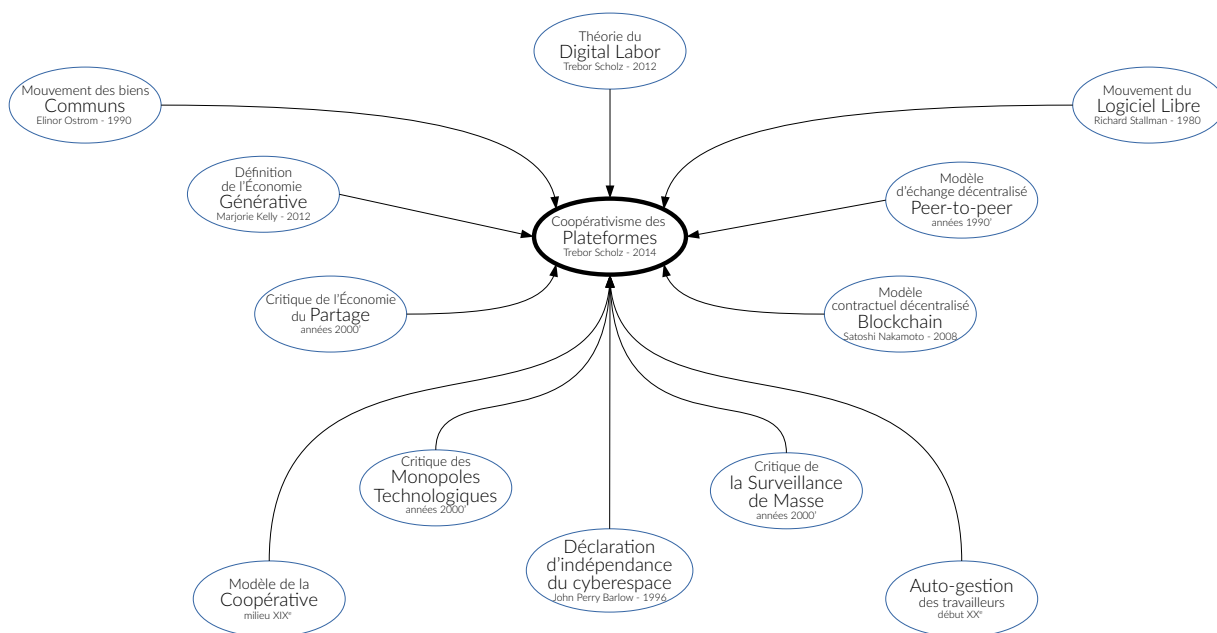


FIGURE 1.6 – Origines du coopérativisme des plateformes

must be able to avoid the coerced and undisclosed surrender of personal information and digital identity to third-parties who may or may not be reliable stewards of such information. »²² (5) Aussi, le mouvement du coopérativisme des plateformes embrasse une dynamique plus large de contestation contre l'idéologie et les pratiques « colonialistes »²³ colportées par la Silicon Valley, ou d'une manière générale par le « capitalisme des plateformes » (*platform capitalism*).

1.5.4 Un mouvement des biens communs

We are in the cultural moment of cooperation. Wikipedia, free and open-source software, citizen journalism, and other forms of commons-based peer production have made normal people encounter cooperation and its products as a matter of everyday practice.

— Y. BENKLER [4]

Comme mentionné précédemment, de nombreuses influences ont participé à l'émergence de ce mouvement (Figure 1.6). Mais parmi celles-ci, le mouvement des biens communs est

22. Nous devons empêcher la capture forcée et discrète des informations personnelles et de l'identité numérique par des tierces parties qui pourraient être — ou pas — des gestionnaires fiables de ces informations. — traduction libre

23. Ce « colonialisme » a été mentionné précédemment. Il prend de nombreuses formes, et fait des utilisateurs des sources de revenus inépuisables. Conquête de l'attention, intrusion dans l'intimité, « Colonisation par l'amour » [49] sont autant de manifestations de cet esprit de conquête obscènement invasif et débridé.

probablement le plus important. Ce dernier prend racine dans les études menées par OSTROM²⁴ en 1990, qui font l'analyse de la gestion institutionnelle des ressources communes comme l'eau et les zones de pêches. Ces éléments naturels, pour ne pas être épuisés ou corrompus, nécessitent une gestion collective, au risque sinon de voir se réaliser la « tragédie des communs » [32]. C'est le célèbre « dilemme du prisonnier » de la théorie des jeux, dont les protagonistes vont se trahir parce qu'ils ne peuvent pas communiquer, alors qu'ils auraient tout intérêt à coopérer pour limiter leur peine. De ces analyses, OSTROM dresse un constat mitigé :

« What one can observe in the world, however, is that neither the state nor the market is uniformly successful in enabling individuals to sustain long-term, productive use of natural resource systems. »²⁵ (58, page 1)

Dans son ouvrage, elle essaie d'identifier, à travers une méta-analyse, les éléments discriminants qui font le succès ou l'échec d'une gestion collective de ces ressources limitées.

Le modèle des communs se situe « beyond states and market »²⁶ (58, page 2) ; vers l'auto-gestion. Resonate.is est par exemple construit à partir de chaînes de blocs et son moto est « everyone owns it, musicians, fans and people who build it. » Sur sa page d'accueil, l'application Resonate.is s'adresse aux auditeurs, aux musiciens, aux maisons de disques, aux contributeurs — développeurs — et aux investisseurs en leur offrant la même opportunité de posséder et de gouverner. Un rapide formulaire permet de se joindre à l'aventure. La structure du pouvoir est profondément changée, puisqu'elle est intégralement distribuée entre la multitude d'acteurs qui peuvent interagir avec l'outil. Comprenons donc le chamboulement, le tremblement de terre qu'un tel modèle pourrait provoquer s'il se démocratise. La démocratie ne se conduit pas au sommet d'une pyramide. Pour prospérer, ses racines doivent s'étendre dans chaque interstice de la société.

1.6 L'ingénierie des exigences

Many systems have failed – often very expansively or even disastrously – because their requirements were not properly determined.

— M. JACKSON [40, page 2]

24. Récipiendaire du prix Nobel d'économie en 2009.

25. *Ce que l'on peut observer dans le monde, cependant, c'est que ni l'État ni le marché ne parviennent de manière uniforme à soutenir l'utilisation à long terme et efficiente des systèmes de ressources naturelles.* — traduction libre

26. *au delà des États et du marché* — traduction libre

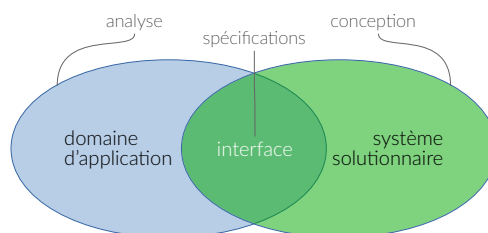


FIGURE 1.7 – Vue d'ensemble sur l'ingénierie des exigences

Inspiré de : BRAY, I. K., 2002, page 10 [7]

1.6.1 Principes saillants

L'ingénierie des exigences (Figure 1.7) est une discipline visant à étudier l'ensemble de ce qui est requis pour apporter des solutions à des problèmes identifiés dans un **domaine d'application** ou domaine du problème (*problem domain*). Ce dernier correspond à une portion du réel concernée par la problématique posée. Cette discipline vise à transformer des problèmes flous, vaguement définis et ouverts, en des problèmes clairement formulés dont on aura levé toute ambiguïté d'interprétation.

En d'autres termes, elle vise à établir avec rigueur et précision *ce que* le système doit faire, *où* va-t-il le faire et non pas *comment* va-t-il y parvenir, ce dernier point étant couvert par la conception interne (*internal design*). Elle vise donc à identifier et qualifier le *problème*.

Parcours des concepts

Deux étapes peuvent être identifiées dans cette discipline. La première – l'analyse – consiste à définir le **domaine d'application** et la seconde – la spécification – vise à décrire l'interface, c'est-à-dire la surface de rencontre entre le domaine d'application et le système-solution.

Les exigences en tant que telles sont définies ainsi par BRAY : « the effects that the client wishes to be brought about in the problem domain » ²⁷ (7, page 14) . Il a répertorié trois types d'exigences (Figure 1.8).

- Les contraintes commerciales. Elles limitent le coût associé au développement et à la maintenance du système-solution.
- Les contraintes de conception. Elles réduisent les choix technologiques pour résoudre *comment* la solution va fonctionner.
- Les exigences fonctionnelles : c'est l'ensemble des comportements du système ayant des effets dans le **domaine du problème**.

27. les effets que le client souhaite constater dans le domaine du problème — traduction libre

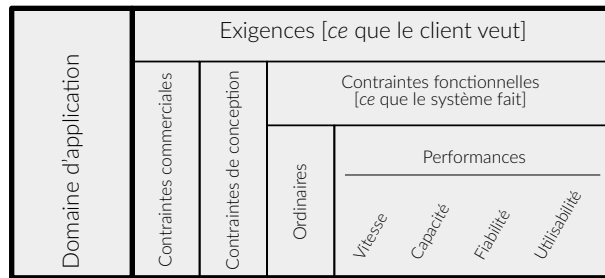


FIGURE 1.8 – Typologie des exigences

Inspiré de : BRAY, I. K., 2002, page 20 [7]

Parmi les exigences fonctionnelles, il distingue deux types :

- Les exigences fonctionnelles ordinaires, qui sont au cœur du cahier des charges. Elles décrivent *ce que* le système fait.
- Les contraintes de performance qui sont des attributs des exigences fonctionnelles. Elles peuvent concerner des éléments de vitesse, de capacité, de fiabilité ou encore d'utilisabilité²⁸. Elles décrivent *le périmètre acceptable* de ce que le système fait.

Les cinq phases

L'ingénierie des exigences se décompose en cinq phases :

1. l'exploration (Figure 1.9, item B) vise à acquérir des informations brutes sur le **domaine du problème**. Et ce depuis la littérature, les individus qui en sont les experts, les (futurs) utilisateurs ou autres personnes (Figure 1.9, item A). À l'issue de ce processus sont produites les notes d'exploration (Figure 1.9, item G) ;
2. l'analyse (Figure 1.9, item C) a pour but de synthétiser les notes d'exploration pour agréger en connaissances structurées des données éparses. Elle vise à documenter le **domaine du problème** ainsi qu'à identifier les exigences (les besoins avérés) qui en émergent. Son produit est le cahier des charges (Figure 1.9, item H) ;
3. la spécification (Figure 1.9, item D) consiste à inventer et à formuler le comportement de la solution pour qu'elle ait les effets désirés sur le **domaine du problème**, tels que décrits dans la section « exigences » du cahier des charges. Dans le contexte d'un contrat, le document qui en résulte (Figure 1.9, item I) est d'une importance capitale car il structure l'entente entre client et prestataire ;

28. Au sens ergonomique.

4. la spécification d'interfaces homme-machine (**Figure 1.9, item E**) vise à identifier les interactions nécessaires entre l'utilisateur et la machine, sans pour autant se supplanter au travail de concepteur d'interface homme-machine qui pourra faire le choix, si la liberté lui est donnée, de proposer ses propres solutions à partir des interactions identifiées. Le produit de cette phase est le document de spécifications **Interface Homme-Machine (IHM)** (**Figure 1.9, item J**);
5. la validation consiste à la relecture et au croisement des informations tirées des parties précédentes, avec l'accompagnement du client. Elle peut donc consister à valider les besoins, contrôler les exigences ou encore confirmer la description du domaine d'application.

Ces « phases » ne sont pas réalisées de façon linéaire, mais forment un cycle d'itérations jusqu'à ce qu'un raffinement suffisant soit atteint [7, page 48].

1.6.2 Prototypage

Le prototypage est d'usage dans l'ingénierie des exigences. Bien entendu, il n'a d'utilité que s'il est utilisé par des utilisateurs (intervenants) qui ont une expertise du domaine d'application. Il peut notamment être utilisé comme vecteur d'exploration pour mettre en évidence certaines caractéristiques du **domaine d'application**, ou encore comme outil de validation. Parmi les avantages incontestables, on peut citer :

1. la possibilité de confronter la représentation de l'analyste et de l'équipe de développement avec l'expertise des parties prenantes ;
2. la possibilité pour les parties prenantes de prendre conscience, en les mettant en oeuvre par l'intermédiaire du prototype, de certaines connaissances et savoir-faire qu'ils trouvaient trop évidents pour sentir le besoin de les vulgariser.

BRAY propose la classification suivante [7, page 242] :

- **prototype exploratoire** : assez sommaire et incomplet, il sert principalement à l'exploration ;
- **prototype définitif** : il participe aux spécifications comportementales ;
- **prototype structurel** : particulièrement utile pour la conception interne, il peut aussi être utilisé en ingénierie des exigences pour tester la faisabilité de l'outil et valider l'utilité et l'utilisabilité de certaines fonctionnalités ;
- **prototype évolutif** : en général, hors du champ de l'ingénierie des exigences, son aboutissement est le produit final.

1.6.3 Le langage univoque

EVANS fait la promotion d'un langage omniprésent (*ubiquitous language*) – parfois traduit par « langage univoque » – pour réconcilier des disciplines qui ont chacune leur jargon, en levant toutes les ambiguïtés inhérentes à chaque domaine, et en distinguant plusieurs définitions d'un même mot ne revêtant pas la même signification d'un domaine à l'autre : « To create a supple, knowledge-rich design calls for a versatile, shared team language, and a lively experimentation with language that seldom happens on software projects. » ²⁹ (19, page 42)

Parfois, les experts d'un domaine ont des connaissances qu'ils n'ont jamais verbalisées, parce qu'ils les considèrent comme trop évidentes, alors qu'elles ne le sont pas pour les développeurs. Réciproquement, un développeur qui discute avec un expert de domaine doit apprendre à traduire des relations entre artefacts de programmation (par ex., classe, objet, table et colonne SQL, instance, API) en des termes qui appartiennent au domaine d'expertise.

Cette démarche est déjà évidente pour les chercheurs habitués aux expériences transdisciplinaires, mais EVANS va beaucoup plus loin en décrivant les implications sur le travail d'analyse et de conception de l'élaboration d'un dictionnaire de domaine univoque. Un tel dictionnaire est un excellent support d'exploration et d'analyse, et il a été adopté dans le présent projet (*dictionnaire de domaine univoque*).

Il mentionne deux implications majeures. Tout d'abord, le travail de *modélisation analytique* n'est plus neutre vis-à-vis du domaine du problème. Car EVANS prescrit de lever les incohérences, les ambiguïtés et les imprécisions qui émergent du *vocabulaire métier* (le *jargon*), et donc potentiellement d'« altérer » le *domaine du problème*. La deuxième implication importante est que la modification d'un terme du dictionnaire revient à modifier la définition du *domaine du problème*, ce qui nécessite une révision du *modèle analytique*. Par ricochet, les spécifications ainsi que le logiciel doivent être amendés pour refléter la nouvelle sémantique. Autrement, une désynchronisation entre le modèle analytique et le *modèle analogique* a lieu, ce qui peut compromettre le projet.

29. Pour parvenir à une expérience de conception enrichissante et souple, il faut développer un langage d'équipe polyvalent et expérimenter avec le langage, ce qui se produit rarement dans les projets de développement logiciel.
— traduction libre

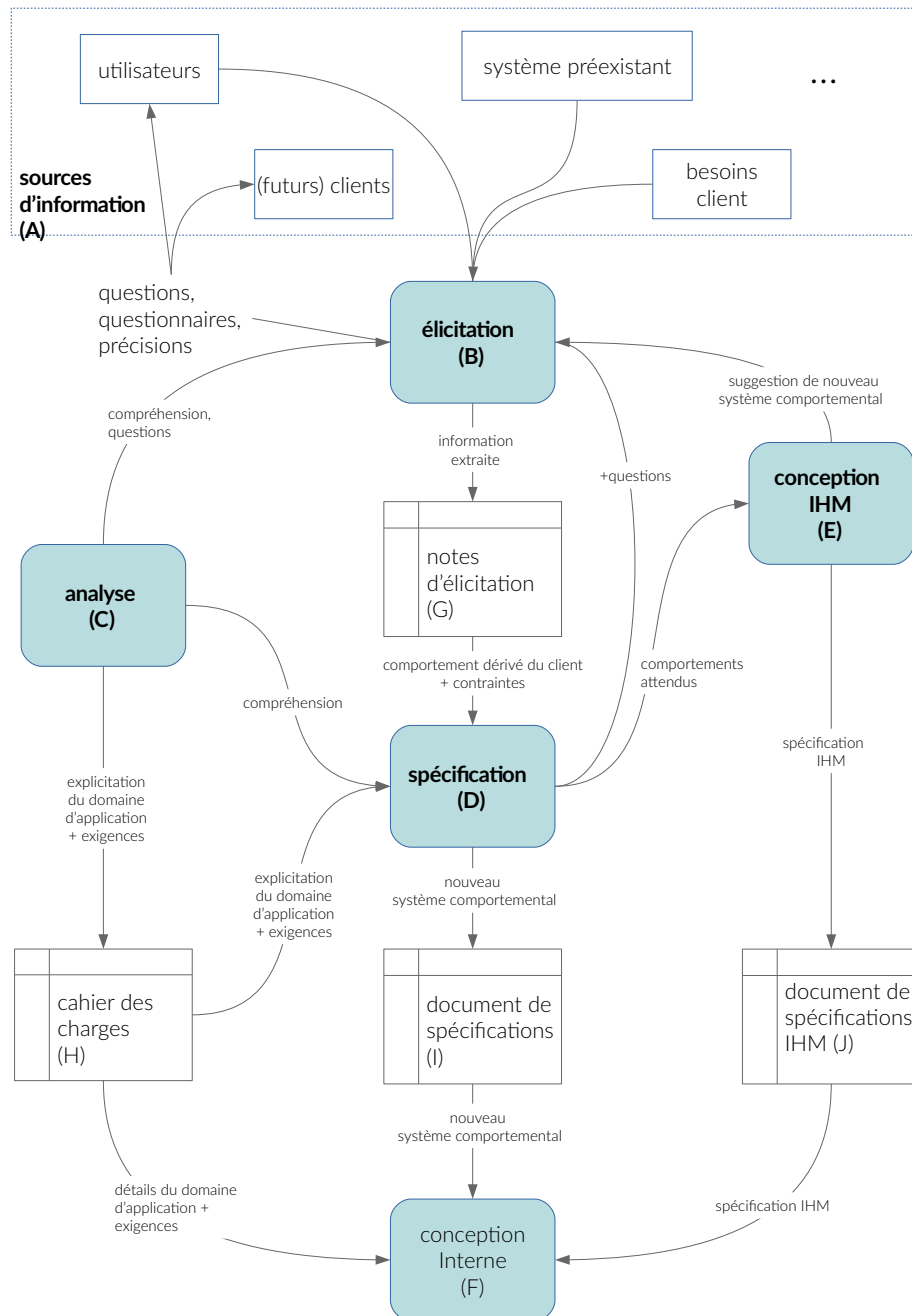


FIGURE 1.9 – Modèle de processus en ingénierie des exigences

Traduit de l'anglais : BRAY, I. K., 2002, page 31 [7]

Chapitre 2

Objectifs et méthodologie

Ce chapitre définit les objectifs de recherche, et les caractéristiques fondamentales que la plateforme collaborative COLAB devra vérifier. La méthodologie et les techniques utilisées y sont détaillées pour donner des clés de compréhension de la démarche globale. Dans la [section 2.1](#), les objectifs de recherches sont exposés. Ensuite, on propose le contexte d’exploration au sens de l’ingénierie des exigences ([section 2.2](#)). Le modèle de flux de travail l’accompagne pour décrire le déroulement des étapes de travail ([section 2.3](#)). Dans la [section 2.4](#), on présente les productions qui vont être réalisées dans le cadre de cette étude. On explique dans la [section 2.5](#) la méthodologie utilisée pour réaliser des tests d’utilisabilité. Enfin, on développe la méthode d’analyse dans la [section 2.6](#).

2.1 Identification des Objectifs

Plusieurs disciplines nécessitent une analyse *a posteriori* du comportement d'individus mis en situation dans un cadre clinique ou expérimental. Dans le cas qui nous concerne, des **analystes** doivent visionner des vidéos d'expérimentation en appartement. Ils sont secondés par des **opérateurs en domotique** qui sont en charge de l'acquisition et la validation de données d'expérimentation, et peuvent signaler d'éventuels dysfonctionnements. Aussi, le besoin d'une infrastructure permettant la collaboration entre ces expertises est manifeste.

2.1.1 Contexte

Auparavant, les participants ont complété une expérimentation, et différentes caméras ont capté les scènes. En plus des évaluateurs, d'autres chercheurs peuvent avoir besoin d'accéder aux captures vidéo, notamment des statisticiens qui travaillent en parallèle des ergothérapeutes. Dans le cadre du projet de recherche à l'origine de COLAB, le projet **DEI**, les étudiants et chercheurs nécessitant cet accès sont issus de plusieurs institutions, à savoir l'Université de Sherbrooke et l'Université de Montréal. Il est aussi à noter que les participants ont donné leur accord pour que des fichiers audiovisuels soient capturés en signant un formulaire de consentement. Ce formulaire stipule : « Par ailleurs, avec votre consentement, ces enregistrements audiovisuels pourraient être utilisés à des fins d'étude, d'enseignement, de recherche ou lors de conférences scientifiques. » (23, page 2)

2.1.2 Protocole de recherche du projet pilote DEI

Le protocole de recherche du projet **DEI** formule ainsi les objectifs visés par le groupe :

« 1) Documenter et comparer les comportements de participants avec et sans troubles cognitifs lors de la réalisation d'AIVQ en laboratoire ; 2) Recueillir des mesures objectives sur la réalisation des **AIVQ** par le biais de capteurs ; 3) Déterminer les liens potentiels entre les comportements observés et ceux recueillis par les capteurs ; 4) Vérifier la faisabilité de réaliser une expérimentation en multi-sites, c.-à-d. en impliquant deux laboratoires contenant un appartement intelligent. » (15, page 1)

Le logiciel COLAB vise à soutenir les travaux utiles à la réalisation des objectifs n° 3 et 4 du protocole de recherche du projet **DEI**. En effet, une exploitation multi-sites exige l'accès à distance des vidéos, dans des dispositions qui les rendent au mieux exploitables pour l'analyse clinique et statistique. COLAB doit donc fournir les conditions pour qu'une bonne collaboration

TABLEAU 2.1 – Objectifs de recherche

Terme	Description
O.I	<i>Interdisciplinarité.</i> Évaluer comment l’outil peut être conçu pour permettre et encourager la collaboration de différentes expertises autour d’un projet d’étude.
O.II	<i>Généricité.</i> Évaluer dans quelle mesure cet outil peut être rendu générique, eu égard à l’objectif n° 4 du projet DEI.
O.III	<i>Adaptabilité.</i> Identifier les caractéristiques que doit satisfaire l’environnement d’acquisition pour permettre à un tel logiciel d’être déployé sur différents sites.
O.IV	<i>Gouvernance.</i> Identifier comment cet outil, une fois implémenté, peut progresser et perdurer dans le milieu académique pour faciliter la collaboration entre chercheurs.

interdisciplinaire survienne, en offrant d’améliorer la qualité, l’efficience et l’interpénétration des expertises.

2.1.3 Objectifs de recherche associés à COLAB

Les objectifs de recherche associés au prototypage et à la spécification de COLAB sont donc dans la prolongation de ceux du projet DEI (section 2.1).

La nature pluridisciplinaire de la collaboration autour de l’outil est un élément central de COLAB (O.I). À terme, ce dernier devrait faciliter la transdisciplinarité, c’est-à-dire l’union innovante de disciplines qui chacune se dépasse et se redéfinit pour aboutir à de nouvelles pratiques.

Ensuite, la volonté — exprimée dans le protocole de recherche — d’évaluer la faisabilité d’une expérimentation multi-sites a influencé la perspective des objectifs O.II et O.III. En effet, la mise en place d’une expérimentation multi-sites nécessite la définition de critères d’homogénéisation techniques, pour évacuer le risque d’une trop grande disparité des données qui rendrait difficile leur exploitation *a posteriori*.

Enfin, l’O.IV trouve son origine dans le besoin d’assurer la vie du logiciel dans un contexte académique qui ne se prête pas toujours au déploiement et au maintien de logiciels, les finance-

TABLEAU 2.2 – Caractéristiques fondamentales de COLAB

Terme	Description
C.I	<i>Accessible.</i> L’outil doit être une application internet riche, compatible avec des postes de travail de résolution moyenne et grande. Sa distribution doit permettre un accès à distance pour une collaboration multi-sites.
C.II	<i>Confidentiel.</i> L’outil doit garantir une excellente confidentialité des participants, en regard de la charte de confidentialité signée par eux. Par conséquent, l’accès à l’outil doit être sécurisé et différencié selon les responsabilités des chercheurs.
C.III	<i>Réflexif.</i> L’outil doit être conçu pour permettre aux utilisateurs de retracer intégralement l’évaluation d’un participant par un évaluateur. C’est-à-dire, rendre l’évaluation transparente en laissant apparaître toutes les observations intermédiaires qui ont abouti aux conclusions de l’évaluateur. Cette caractéristique permet la traçabilité des résultats de recherche. L’outil devient un moyen de <i>réflexivité</i> dans la pratique scientifique.
C.IV	<i>Libre.</i> Le logiciel qui résultera de ce travail doit être <i>libre</i> et donc, <i>open source</i> , pour faciliter le partage, l’innovation et la maintenance par la communauté de chercheurs.

ments étant traditionnellement structurés pour la recherche plutôt que pour les services.

2.1.4 Caractéristiques fondamentales vérifiées par COLAB

Les caractéristiques fondamentales que doit vérifier COLAB sont des critères, évidents pour certains, arbitraires pour d’autres, que devra satisfaire l’application compte tenu du contexte exposé précédemment (section 2.2).

L’accessibilité (C.I) et la confidentialité (C.II) sont des propriétés essentielles. L’accessibilité de l’outil vient naturellement en partant de la contrainte multi-sites. Les collaborateurs doivent pouvoir accéder aux données d’observation depuis n’importe quel lieu de travail. Quant à la confidentialité des données des participants, elle va de soi lorsqu’on tient compte du formulaire de consentement, cité en introduction de cette section.

La caractéristique réflexive (C.III) découle du souhait de faire de COLAB un outil qui non seulement permet, mais « augmente » le travail d’analyse *a posteriori*. Aussi, elle accompagne

l’objectif **O.I** en laissant transparente la méthodologie qui aboutit à ses résultats, en permettant potentiellement une meilleure interpénétration des disciplines coalisées. L’hypothèse est faite qu’autoriser une certaine réflexivité dans la tâche d’évaluation ne peut qu’améliorer la qualité de travail produit. Elle reste, dans le cadre de ce mémoire, une hypothèse difficilement vérifiable, mais on reviendra plus tard sur les indices qui participent à soutenir sa pertinence.

Enfin, la caractéristique *libre* (**C.IV**) a été établie de façon partiellement arbitraire, dans la mesure où elle est motivée par un idéal universitaire¹. Pour autant, elle n’est pas totalement déconnectée des objectifs de recherche, notamment l’**O.IV** et le souhait de voir cette solution se concrétiser et perdurer. Le développement open source d’un logiciel — dans un contexte *libre* ou pas — offre de nombreux avantages qui font augmenter ses chances de se pérenniser. Parmi ceux-ci, on peut citer la fiabilité, la stabilité, le coût, la flexibilité, l’auditabilité et la liberté [6].

2.1.5 Épilogue

L’ingénierie des exigences va constituer le cœur méthodologique pour aborder les différents objectifs et caractéristiques énumérés précédemment. Une brève description de cette discipline est faite dans la **section 1.6**. Dans la section qui suit est donné le contexte du présent travail.

TABLEAU 2.3 – Liste des parties prenantes

Partie prenante	Apports et connaissances extraites
Administrateur système du laboratoire	Expertise domotique sur l’environnement d’acquisition de l’appartement « intelligent » du laboratoire DOMUS , c’est-à-dire une partie de l’environnement opératoire.
Chercheurs associés et étudiants en ergothérapie, évaluateurs	Retours sur l’utilisation du prototype, expertise sur le domaine métier , co-conception, évaluation de l’utilisabilité.
Étudiants en informatique, utilisateurs	Retours sur l’utilisation du prototype, évaluation de l’utilisabilité et de l’utilité.

1. Certains étudiants et chercheurs, dont je fais partie, considèrent le logiciel libre comme une évidence dans le monde académique (E. von HIPPEL et G. von KROGH [38]). C’est pour nous, premiers bénéficiaires des savoirs humains, un impératif moral que de livrer sans condition le produit de nos recherches à la communauté, sous forme manuscrite ou logicielle.

2.2 Contexte d'exploration

2.2.1 Parties prenantes

Différentes parties prenantes ([Tableau 2.3](#)) doivent être identifiées avant même de décrire le **domaine d'application**. Ces individus sont issus de l'environnement dans lequel s'inscrit le projet pilote **DEI**.

2.2.2 Sources d'exploration

Par « source » ([Tableau 2.4](#)), il faut comprendre des sources qui vont nourrir les productions associées au présent mémoire. Les productions sont énumérées dans la [section 2.4](#). Les sources apparaissent de manière sommaire dans la figure 2.2 de la [section 2.3](#). On peut distinguer deux types de sources d'exploration. Les sources référencées dans un travail de recherche fournies au chapitre « état de l'art » du présent mémoire ([chapitre 1](#)), et les sources externes, qui sont pour la plupart des documents à l'usage du projet **DEI**, non publiés.

TABLEAU 2.4 – Inventaire du matériel d'exploration et d'analyse

Domaine	Document	Description	Réf.
Domaine métier	Méthodologie d'évaluation	Description d'une méthode employée pour l'analyse <i>a posteriori</i> , par visionnement des vidéos d'expérimentations (Figure 2.1, étape 3.A). L'expertise principale est l'ergothérapie.	n/c
	Grille d'évaluation	Ensemble des critères définis par les évaluateurs pour mesurer l'autonomie des participants (Figure 2.1, étape 3.A).	n/c
	Protocole de recherche du projet DEI	Document constitutif des axes de recherche définis pour le projet pilote DEI.	[15]
	Formulaire d'information et de consentement	Document contractuel garantissant le respect de la confidentialité des participants aux expérimentations du projet DEI.	n/c
	Revue de littérature sur les techniques d'évaluation par AIVQ	Littérature universitaire qui traite des méthodes pour évaluer l'autonomie des sujets dans le contexte des AIVQ.	1.3
Environnement opératoire	Documentation de l'environnement d'acquisition	Documentation interne au laboratoire DOMUS.	n/c
Domaine du problème	Veille technologique des solutions logicielles voisines	Inventaire de solutions logicielles similaires à COLAB.	1.2
	Introduction aux concepts voisins de l'interdisciplinarité	Description de la terminologie propre au croisement des disciplines.	1.1
	Revue de littérature sur le coopérativisme des plateformes	Rétrospective sur l'histoire récente du mouvement civique du coopérativisme des plateformes.	1.5

2.3 Modèle de flux de travail

2.3.1 Cadre de travail

Comme cela a été rappelé dans la [section 1.6](#), l’exploration et l’analyse sont des étapes préalables pour établir une première version des spécifications. L’ingénierie des exigences n’est pas une méthodologie exhaustive, et son application doit se faire avec discernement et pragmatisme, en prenant compte des contraintes d’exécution relatives au groupe de travail et au projet auquel l’ouvrage est rattaché.

Ce travail a lieu dans un cadre de recherche : il n’est donc pas directement motivé par un contrat liant un client et un prestataire. Les exigences – normalement définies comme une formalisation des besoins du client – constituent ici un objet de recherche à part entière, couvertes par les objectifs [O.I](#), [O.II](#) et [O.III](#).

Dans une première étape d’élaboration, un besoin clairement identifié est urgent à satisfaire qui justifie la conception et l’implémentation rapide d’un [prototype structurel](#). Ce prototype va permettre d’agir comme un vecteur d’exploration auprès des utilisateurs de l’application, qui sont représentatifs du profil d’utilisateur final. Ainsi, par étapes successives et divers raffinements, une application pourra être spécifiée pour une prochaine itération du protocole de recherche du projet [DEI](#).

2.3.2 Planification

Pour tenir compte des contraintes d’exécution propres au projet [DEI](#), le processus de spécification se fait donc en deux temps ([Figure 2.2](#), *cycles 1 & 2*). Le projet [DEI](#) est structuré en quatre étapes ([Figure 2.1](#)).

L’exploitation clinique des données de l’étape 3.A nécessite l’accès par les évaluateurs aux vidéos d’expérimentation acquises à l’étape 2. Ce besoin est clairement établi en amont et justifie la création d’un prototype (COLAB 0.4) qui précédera une version plus aboutie, incluant une étude des besoins de ces analystes en relation avec le [domaine d’application](#), dans lequel ils évoluent.

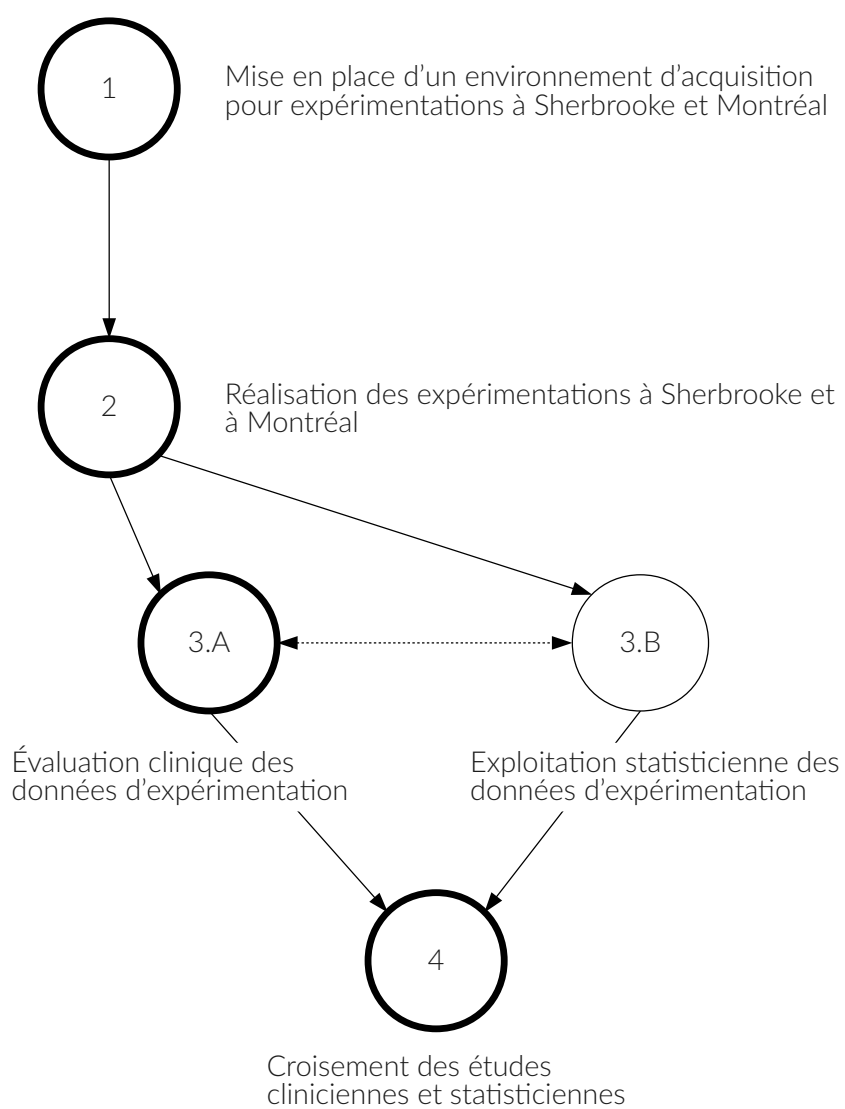


FIGURE 2.1 – Étapes du projet DEI

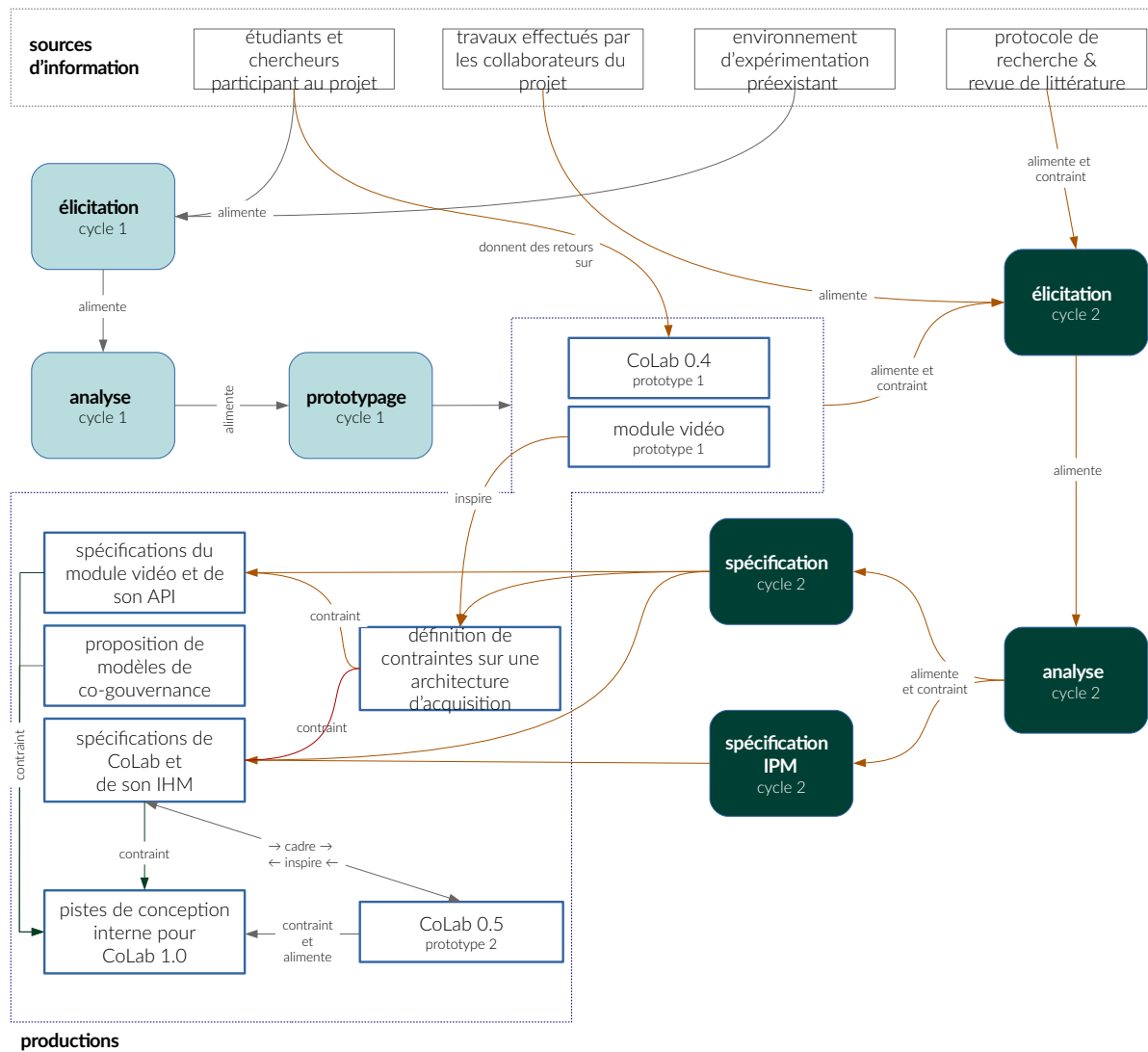


FIGURE 2.2 – Planification du flux de travail

2.4 Caractérisation des productions

Dans cette section sont détaillées les *productions* visibles dans la figure 2.2.

2.4.1 Productions logicielles (prototypes)

Dans ce projet, des prototypes exploratoires et structurels sont développés (Tableau 2.5). Ces prototypes apparaissent dans la Figure 2.2 (COLAB 0.4, 0.5 & module vidéo), à l'exception de *ipcamsh* dont la réalisation a été jugée trop précipitée pour faire partie du processus d'ingénierie des exigences. On le considère plutôt comme un élément du *domaine du problème*.

2.4.2 Productions documentaires

Ces éléments textuels (Tableau 2.6), développés dans le chapitre 5 du présent mémoire, sont mis en relation avec les objectifs de recherche (section 2.1).

TABLEAU 2.6 – Inventaire des productions documentaires

Document	Description	Réf.
Définition de contraintes sur une architecture d'acquisition	En réponse aux objectifs O.II et O.III (généricité et environnement), la description d'une interface de programmation modulaire la plus générique possible. Cette interface fait la médiation entre COLAB et un environnement d'acquisition quelconque qui respecte un ensemble prescrit de critères.	chap. Spécifications
Proposition de modèles de co-gouvernance	En réponse à l'O.IV, description d'un modèle de gouvernance qui permet à plusieurs universités de co-détenir le logiciel pour assurer sa pérennité.	chap. Spécifications
Spécification de COLAB et de son IHM	Conception externe de l'application COLAB et de son IHM. Respecte scrupuleusement les exigences définies à la fin du chapitre d'analyse.	chap. Spécifications

— suite à la page suivante —

Inventaire des productions documentaires – *suite*

Document	Description	Réf.
Pistes de conception interne pour COLAB 1.0	Éléments de conception interne qui ont émergé des phases de prototypage. Ces pistes ne sont pas péremptoires : l'idée étant plutôt d'assurer la transmission de solutions pertinentes, eu égard aux caractéristiques fondamentales désirées pour l'application.	<i>chap. Spécifications</i>

TABLEAU 2.5 – Inventaire des programmes

Prototype	Cycle	Description
CoLAB 0.4	1	Prototype structurel dont l’objectif est d’évaluer la faisabilité du système-solution.
video-node	1	Prototype exploratoire <i>ad-hoc</i> utilisé pour la mise en ligne sécurisée des vidéos d’expérimentation. Correspond au module vidéo de la figure 2.2. Certes, CoLAB 0.4 & 0.5 ont besoin de ce prototype pour offrir l’accès aux vidéos, mais il est qualifié d’« exploratoire » car pour CoLAB ≥ 1.0 , c’est l’interface de programmation décrite dans « Définition de contraintes sur une architecture d’acquisition » qui sera utilisée.
ipcamsh	1	Prototype exploratoire <i>ad-hoc</i> utilisé pour l’enregistrement des vidéos d’expérimentations, lors de l’étape 1 du projet DEI (Figure 2.1). C’est un script <code>bash</code> qui utilise <code>vlc</code> en interne pour lancer l’acquisition simultanée de six sources vidéo.
CoLAB 0.5	2	Prototype structurel utilisé pour valider certaines pistes de conception interne.

2.5 Mesures ergonomiques

On se référera à la [section 4.2](#) pour connaître le détail de la construction du questionnaire d’utilisabilité.

2.5.1 Approche globale

L’expérience utilisateur est évaluée par deux moyens. D’une part, via des retours informels, textuels ou oraux. C’est alors la démarche d’exploration qui est à l’œuvre, dans le but de raffiner les exigences fonctionnelles. D’autre part, via un formulaire d’utilisation ([Annexe F.1](#)). Ce dernier est structuré en différentes parties, qui sont détaillées dans le ([Tableau 4.1](#)) de la [section 4.2](#). Parmi celles-ci, on peut distinguer plusieurs catégories. L’extraction des domaines d’expertise et des tâches servent à identifier le profil d’utilisation. Ce dernier va permettre de croiser les mesures de pertinence fonctionnelle avec le contexte de travail de l’utilisateur, ses connaissances et ses objectifs.

Mais l’évaluation de l’expérience utilisateur ne se limite pas à l’identification de fonctionnalités pertinentes. La **qualité hédonique**, par exemple, mesure l’expérience utilisateur dans sa

dimension ludique, c'est-à-dire jusqu'à quel point l'utilisateur éprouve du plaisir à utiliser un logiciel [34]. *A contrario*, la **qualité pragmatique** mesure l'expérience utilisateur dans sa facilité à atteindre ses buts (par ex., simplicité, contrôlabilité, convivialité). Dans la littérature, il est observé que cette distinction entre facilité d'utilisation (qualité pragmatique) et plaisir d'utilisation (qualité hédonique) est bien comprise par les utilisateurs [36]. Il a été observé que toutes deux ont un impact équivalent et significatif sur l'adoption d'une solution logicielle [39]. La mesure de ces qualités perçues sera faite avec la méthode *AttrakDiff 2* [35].

2.5.2 Mesure de qualité perçue avec AttrakDiff

Le test AttrakDiff a été proposé par HASSENZAHN, BURMESTER et KOLLER en langue allemande [35]. Il est structuré sous forme de vingt-huit items, chacun étant une paire de mots antonymiques – appelés différenciateurs sémantiques – séparée par sept valeurs possibles, la saisie se faisant sur une échelle de Likert. Ces items sont divisés en quatre groupes de sept, chacun étant associé à un aspect du modèle théorique de HASSENZAHN, BURMESTER et KOLLER. Ces groupes forment des dimensions ou sous-échelles. Le premier est l'échelle de **qualité pragmatique** (QP). Il décrit l'utilisabilité et l'utilité du produit. Le deuxième, l'échelle de **qualité hédonique** – stimulation (QH-S), mesure à quel point le produit provoque du plaisir à l'utilisateur. Le troisième, l'échelle de **qualité hédonique** – identification (QH-I), évalue dans quelle mesure l'utilisateur peut s'identifier à lui. Le dernier, l'échelle d'attractivité globale (ATT), offre une perception globale du produit. La version française, retenue dans le questionnaire (*Annexe F.1*), a été élaborée en 2015 par LALLEMAND et al. [47].

2.6 Méthode d'analyse

De nombreux types d'analyse existent. La méthode privilégiée choisie est l'approche pilotée par problème, popularisée par JACKSON [40]. Bien qu'avant tout destinée aux systèmes embarqués, son cadre est suffisamment souple pour s'exporter dans d'autres domaines. Elle consiste à modéliser de façon détaillée le **domaine du problème**, en le décomposant en un ensemble de sous-problèmes et en identifiant les surfaces nécessitant des spécifications. Dans cette méthode, la prudence est de mise à ne pas confondre la description du problème et des besoins avec l'étape de spécifications, et c'est ce qui fait son intérêt. Nous avons ajouté à cette méthode un développement par itérations successives, comme si l'on labourait le domaine du problème en traçant des sillons. Pour cette dernière, on s'est inspiré de la méthode « *Problem Oriented Software Engineering* » [30]. Cette approche de l'ingénierie des exigences est détaillée dans l'*Annexe C*. En

dehors des notations UML standardisées, les notations utilisées sont répertoriées dans l'[Annexe B](#). Parmi les outils d'analyse, on va retenir :

- *Problem frames* ([Annexe C](#), notation en [Annexe B.3](#));
- dictionnaire de domaine univoque ;
- diagramme de contexte ;
- diagramme d'activité UML (notation en [Annexe B.1](#));
- modèles logiques de données simplifiés (notation « Crow's foot », [Annexe B.2](#)).

Une remarque sur les modèles logiques de données simplifiés : nous avons jugé que dans cette étape précoce d'analyse, il était préférable de s'abstraire des colonnes généralement présentées dans le diagramme logique de données, afin de concentrer l'analyse sur la cardinalité. Aussi, le lecteur pourra se demander à juste titre si ces diagrammes décrivent des structures relationnelles ou objets. Nous n'avons pas tranché, et considérons les entités décrites dans ces diagrammes comme issues d'un niveau d'abstraction métier supérieur aux classes et aux tables.

Chapitre 3

Analyse

Ce chapitre vise à rassembler de façon synthétique les différents aspects saillants du problème, et les effets importants de la solution. Deux descriptions distinctes du problème sont fournies. La première est la description des processus métier et des expertises en dehors du système-solution¹ (section 3.2). On parle alors du « domaine nu ». Ensuite, une « plongée » dans la complexité du problème est réalisée à l’aide des *problem frames* (section 3.3). Puis, une description approfondie de certains sous-domaines est proposée (sections 3.4, 3.5 et 3.6). Enfin, les exigences sont formulées dans les sections 3.7 et 3.8.

1. Cette précision est importante, car certaines analyses en ingénierie des exigences incluent directement le système-solution, comme s’il existait depuis toujours. Nous préférons approfondir la philosophie de JACKSON qui préconise d’explorer le problème loin des interactions avec la machine, « deeper into the world » (40).

3.1 Introduction

L'analyse porte sur deux problèmes distincts mais dépendants. En effet, l'analyse de domaine associée COLAB pourrait être séparée de l'**environnement opératoire** où ont lieu les expérimentations. C'est d'ailleurs l'approche que nous privilégions pour satisfaire les objectifs **O.II** et **O.III**, respectivement *adaptabilité* et *généricité*. L'identification des « caractéristiques que doit satisfaire l'environnement d'acquisition » va aboutir à l'inférence d'exigences de l'interface API SCÉNARII. Dans cette situation, l'environnement d'acquisition existant — dans notre cas l'appartement intelligent du **DOMUS** — ne contraint pas l'élaboration de l'API SCÉNARII. Par contre, notre expérience avec ce dernier doit servir à caractériser une interface qui permet au logiciel COLAB d'extraire les données d'expérimentation nécessaires à leur analyse. On distingue alors :

- l'environnement d'acquisition initial, qui correspond à l'environnement d'acquisition avant l'introduction de l'API SCÉNARII et qui a été utilisé pour le projet **DEI**;
- l'environnement d'acquisition désiré, qui correspond à un environnement d'acquisition compatible avec l'API SCÉNARII et qui respecte ses contraintes environnementales associées.

Le deuxième **domaine du problème** est celui directement impacté par COLAB. C'est un domaine métier articulé autour d'expertises comme l'ergothérapie, la statistique, et dont l'environnement opératoire est physiquement contraint par les modalités d'hébergement et de déploiement d'une application web. On distinguera :

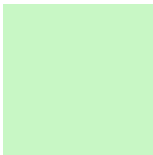




- le domaine métier nu, qui correspond au domaine occupé par les processus métier avant l'arrivée de COLAB (section 3.2);
- le domaine métier enrichi, qui correspond au domaine occupé par les processus métier après l'arrivée de COLAB (sections 3.3, 3.4 et 3.5).

3.2 Description du domaine du problème « nu »

3.2.1 Identification des responsabilités

Nous utilisons « **responsabilité** » à la place d'« acteur », car dans le **domaine du problème** exploré, la hiérarchie organisationnelle est faiblement structurée, c'est-à-dire qu'une grande autonomie est conférée aux différents « acteurs ». Dans les diagrammes qui vont suivre, les couleurs d'actions correspondent à la **responsabilité** qui leur est associée (Tableau 3.1).

TABLEAU 3.1 – Code couleur des responsabilités

Couleur	Responsabilité
	collaborateurs (chercheurs)
	opérateurs d'expérimentation
	opérateurs en domotique
	examineurs-accompagnateur
	chargés de recrutement

3.2.2 Pré-expérimentation

L'équipe de travail procède à diverses tâches pour préparer les expérimentations (Figure 3.1). La première étape consiste à identifier les **classes des participants**. On peut ensuite distinguer deux processus parallèles que sont la recherche de participants d'une part (à gauche), et le rodage du protocole d'expérimentation d'autre part (à droite). Ce rodage a pour but :

- de garantir l'homogénéité entre le déroulement de chaque expérimentation en documentant les **conditions initiales d'expérimentation** ;
- de s'assurer que les capteurs, les micros et les caméras sont disposés d'une manière pertinente par rapport au scénario de travail ;
- de s'assurer que le dispositif d'acquisition (capteurs, micros, caméras) est fonctionnel et

fiable.

L'activité de rodage se termine donc lorsque :

- les conditions initiales d'expérimentation sont stabilisées et validées ;
- la liste de contrôle des conditions initiales est stabilisée ;
- l'acquisition de donnée est stabilisée.

Précisions. Souvent, le processus de recrutement continue bien après la fin de l'activité de rodage du protocole d'expérimentation. À noter aussi que la description d'activité ci-jointe s'affranchit de certaines contraintes de temps. En effet, le rodage du protocole doit en général se faire dans un temps imparti.

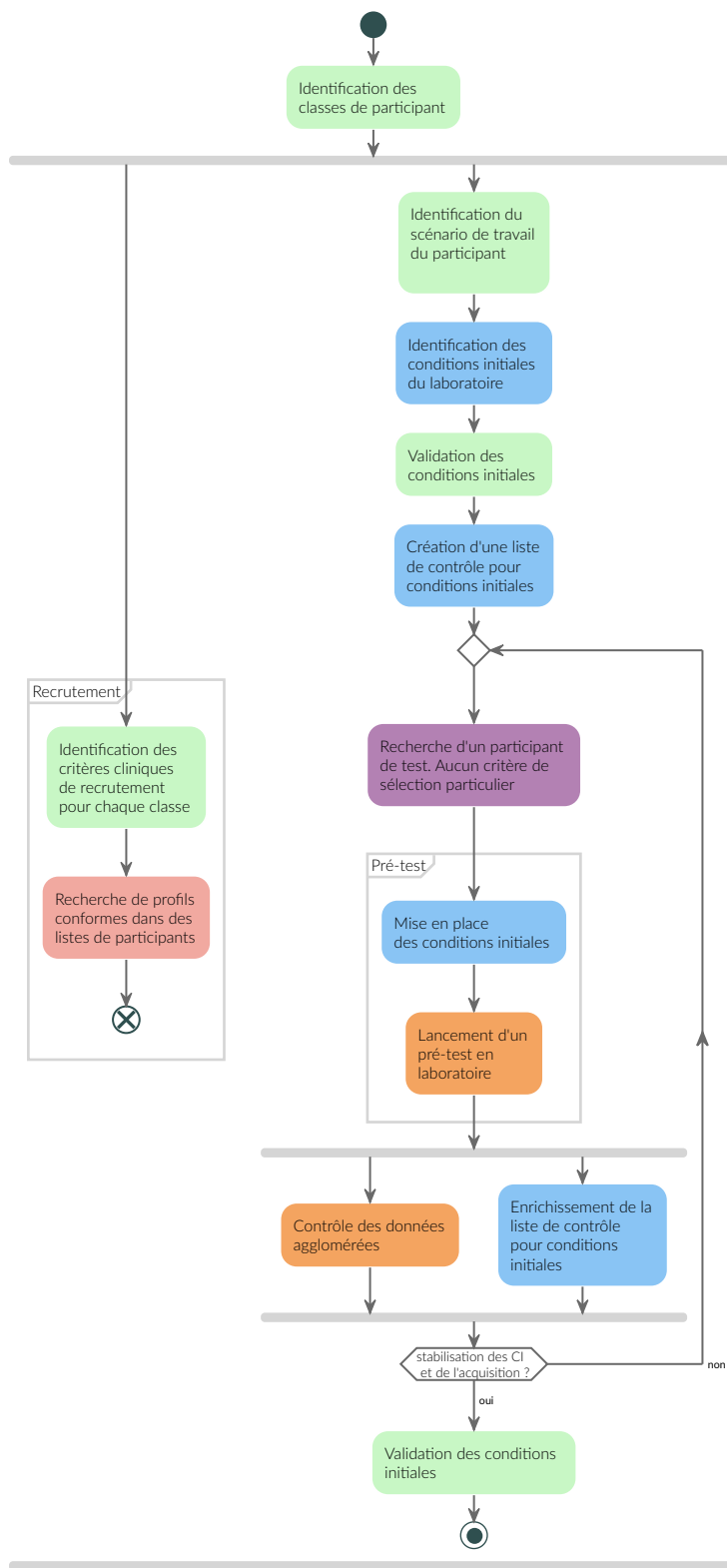


FIGURE 3.1 – Diagramme d'activité pré-experimentation « nu »

3.2.3 L'expérimentation

Le contexte d'expérimentation est donné en Figure 3.2. Un *opérateur en domotique* est en charge du démarrage et de l'arrêt du dispositif d'acquisition. L'opérateur d'expérimentation doit quant à lui garantir la satisfaction des *conditions initiales d'expérimentation* puis il assiste l'*examineur-accompagnateur* en faisant la saisie des tâches réalisées par le *participant*. Elle débute avec une prise de rendez-vous par le *chargé de recrutement* avec le *participant*. L'activité complète de prise d'un rendez-vous pour expérimentation « nue » est décrite dans la Figure 3.3. *Précisions.* En réalité, il existe plusieurs dispositifs d'acquisition (caméras, capteurs filaires, capteurs sans-fil, micro) qui sont démarrés et arrêtés par l'*opérateur en domotique* indépendamment les uns des autres, donc généralement de manière séquentielle. Ils ne sont pas représentés dans ces diagrammes par souci de lisibilité.

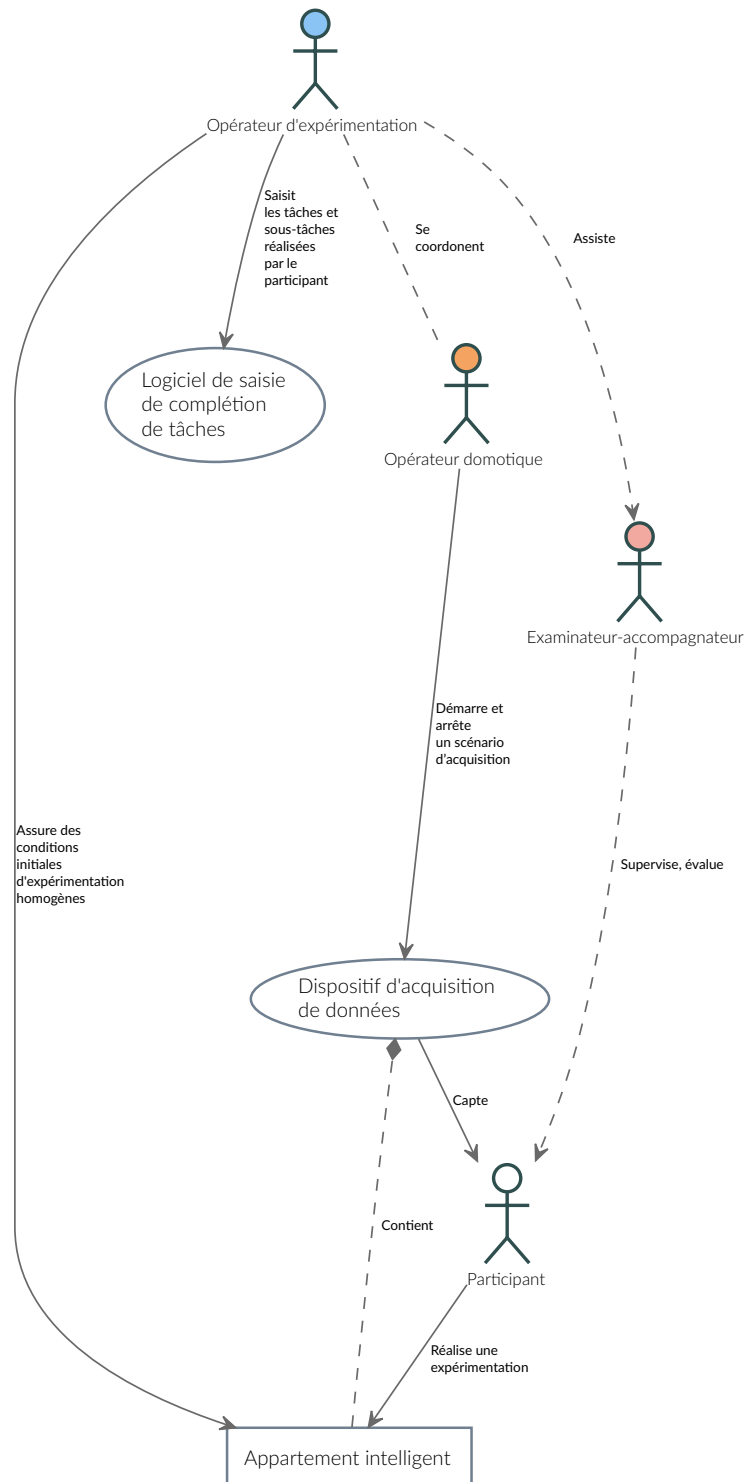


FIGURE 3.2 – Diagramme de contexte pour expérimentation, « nu »

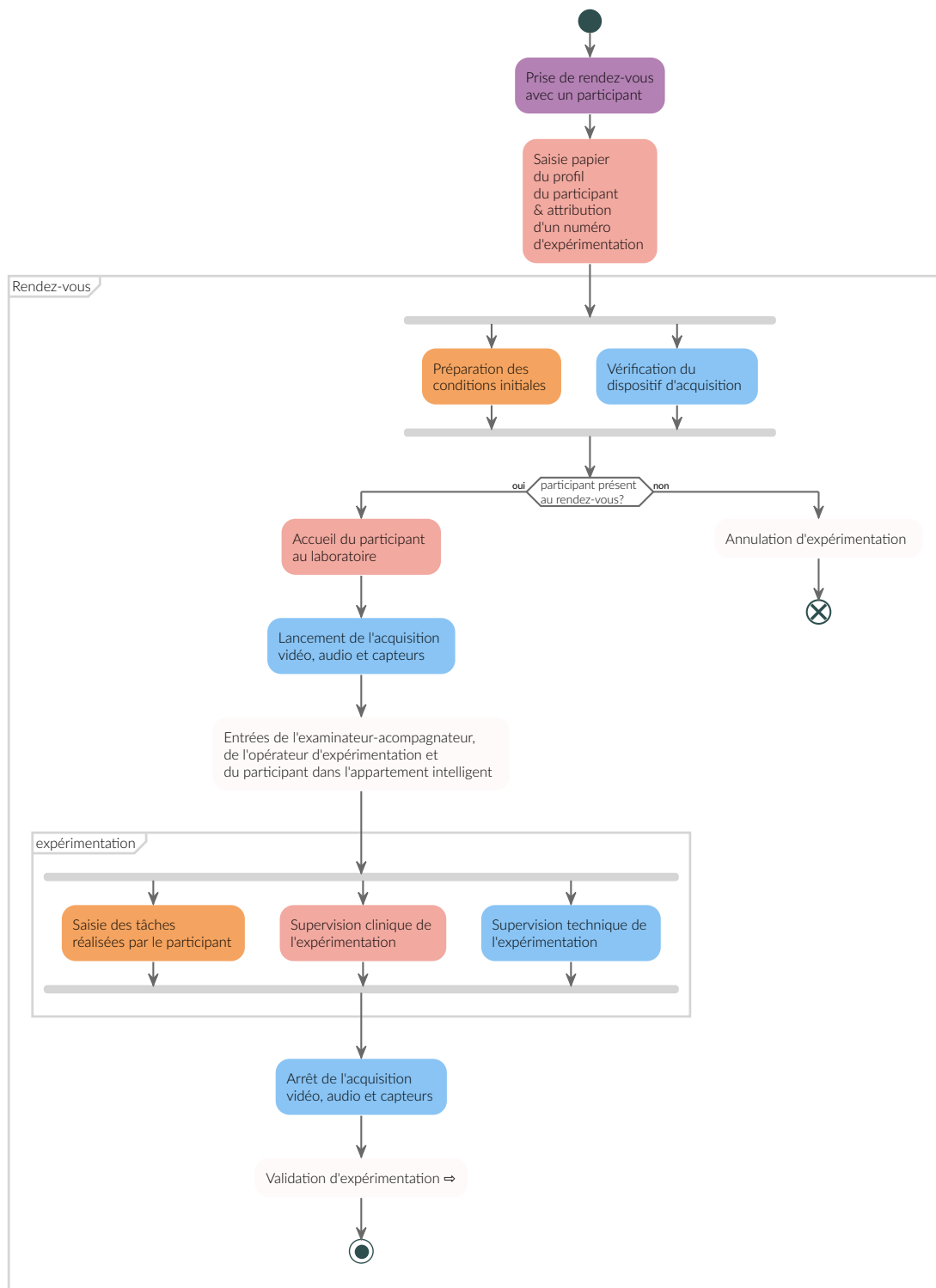


FIGURE 3.3 – Diagramme d'activité pour rendez-vous d'expérimentation, « nu »

3.2.4 Validation d'expérimentation

Les trois ensembles de critères de validation d'une **expérimentation**, visibles en **Figure 3.4**, répondent aux trois questions suivantes :

1. le déroulement de l'**expérimentation** est-il conforme aux critères définis par le groupe de recherche ? C'est *a priori* l'**examineur-accompagnateur** qui est en capacité de le décider, éventuellement en concertation avec les **collaborateurs**.
2. les données récoltées par les capteurs souffrent-elles d'imperfections ou d'irrégularités ? C'est l'**opérateur en domotique** qui peut le constater et le rapporter.
3. les captures vidéos et audios utiles à une analyse future souffrent-elles d'imperfections ou d'irrégularités ? Comme précédemment, l'**opérateur en domotique** peut le constater et le rapporter.

Cette étape est capitale pour que les données d'expérimentation puissent être exploitées dans la phase d'analyse.

3.2.5 L'analyse clinique

Patron de grille

La (**Figure 3.5**) représente un **patron de grille** directement inspiré d'un travail réalisé par des étudiantes collaboratrices du projet **DEI**. On peut observer les feuilles, qui sont des **critères de grille**, et les autres nœuds qui sont des catégories d'évaluation. Le terme « patron de grille » n'existe pas dans le langage courant des experts en ergothérapie. Nous avons dû l'introduire pour distinguer, autour de la **grille d'évaluation**, la structure normative (le patron) et les saisies qui lui sont associées (les **cotations de grille**).

Processus d'évaluation

La **Figure 3.6** décrit le processus métier d'évaluation des **participants** par les **analystes comportementaux**. Les étapes en pointillés sont facultatives, et ne sont pertinentes que lorsqu'une évaluation par pairs est réalisée, c'est-à-dire en groupe de plusieurs évaluateurs. On distingue trois phases dans cette activité. Les deux premières se font à l'aveugle, c'est-à-dire sans connaissance des **classes des participants** :

1. Élaboration du **patron de grille**. La convergence des critères (I.A) consiste, pour un ensemble donné d'**analystes**, à atteindre un consensus sur les composantes de cette grille.

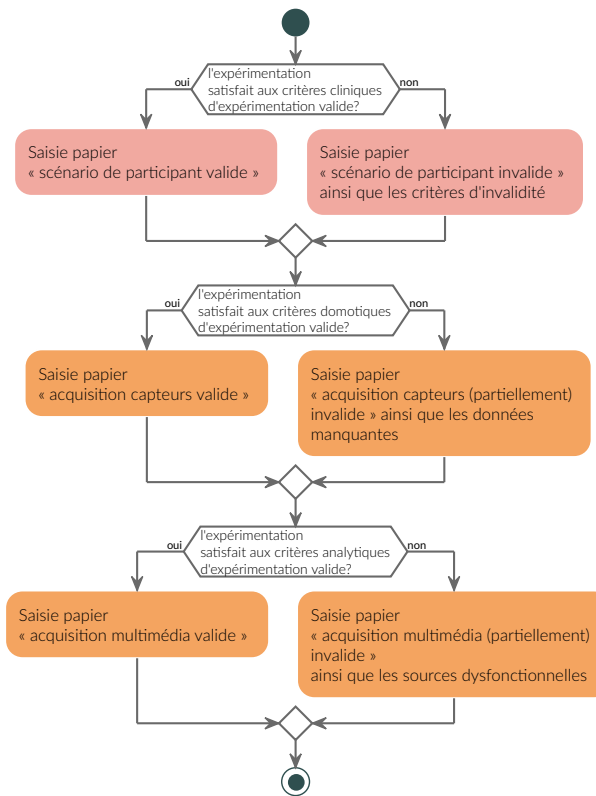


FIGURE 3.4 – Diagramme d’activité pour validation d’expérimentation, « nu »

Lorsque ce consensus est atteint, on parle de stabilisation des critères. En parallèle, le patron de grille est enrichi de nouveaux critères (I.B) à mesure que des **cotations de grille** sont réalisées sur un sous-échantillon des **expérimentations**, jusqu’à saturation de grille.

2. Cotation. Dans cette phase, les **cotations de grille** sont réalisées jusqu’à leur stabilisation (II.B). Un **accord interjuge** peut être mis en place pour évaluer à quel point les évaluations des différents analystes sont disparates ou similaires (II.A).
3. Analyse. Dans cette phase, l’évaluation à l’aveugle est « levée », c’est-à-dire que les analystes accèdent aux **classes des participants** (III.A) et peuvent alors procéder à une analyse statistique ou qualitative de la grille (III.B).

L’évaluation est réalisée à l’aveugle. L’aveugle est un critère important car il garantit que les évaluateurs ne sont influencés :

- ni par l’ordre de venue des participants ;
- ni par la date ;
- ni par la **classe du participant**. Dans notre cas, les trois diagnostics possibles sont « sain », Trouble Cognitif Léger (TCL) et MA.

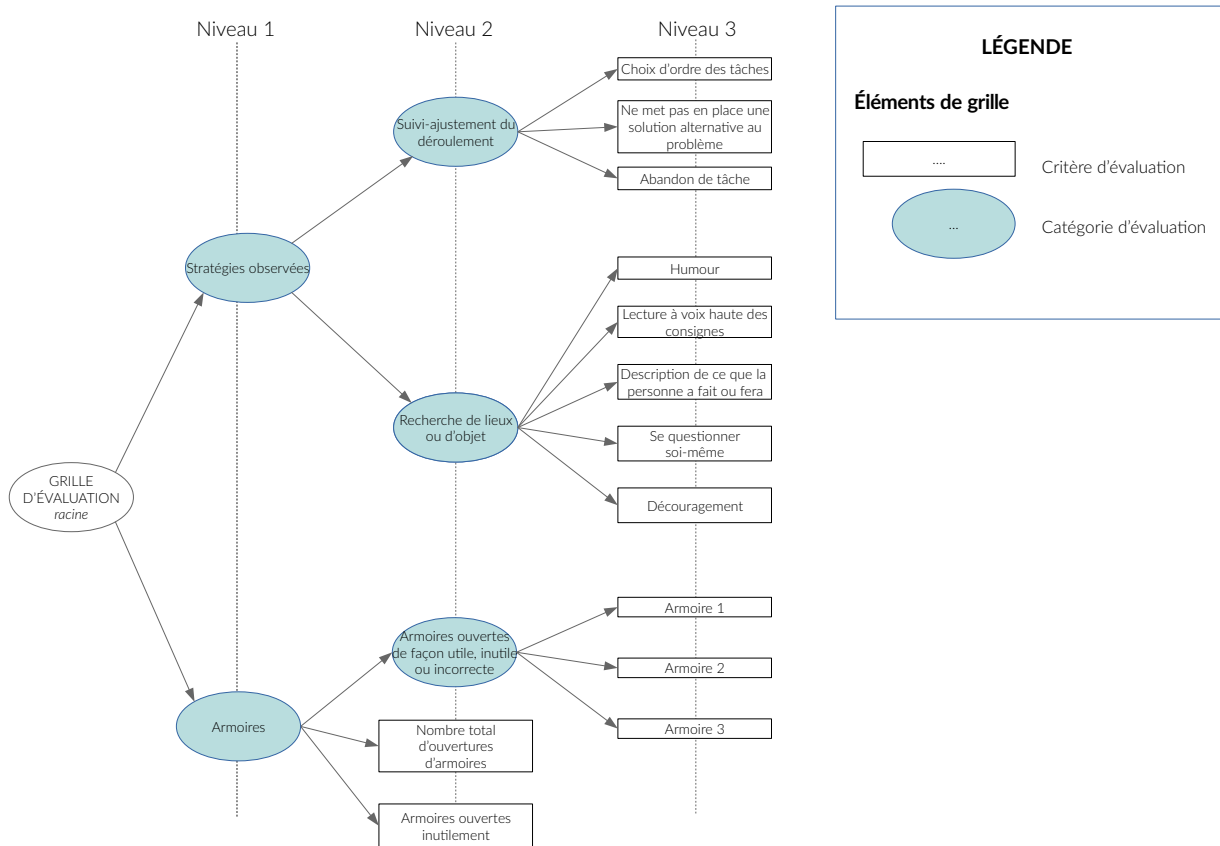


FIGURE 3.5 – Représentation sous forme d'arbre d'un patron de grille

Le critère à l'aveugle n'a pas été, dans un premier temps, implémenté dans le prototype. C'est grâce aux retours des utilisateurs que le besoin est apparu comme impérieux pour satisfaire aux exigences scientifiques de la démarche. À noter aussi que la date n'a pas été cachée pour la phase en aveugle, car elle a été incrustée dans les vidéos. Ce travers ne sera pas répété dans les versions ultérieures du produit, et les contraintes liées à l'aveugle seront clairement exprimées dans la liste d'exigences. Une deuxième protection est l'**accord interjuge**. Selon ce principe, une observation, pour être valide, doit être retenue par deux évaluateurs indépendamment l'un de l'autre.

Précisions. Les informations relatives à l'étape d'évaluation ont été explorées auprès des étudiantes affectées à ce travail, avec confirmation par leur tutrice, collaboratrice au projet **DEI**. Dans cette description du travail d'analyse, nous ne faisons pas intervenir COLAB, et ce, de manière intentionnelle pour isoler le problème de la solution. Autre remarque, les étudiantes **analystes comportementaux** ont visionné les vidéos de six expérimentations pour la phase de convergence des critères. Mais ces critères peuvent grandement varier d'un groupe de travail à

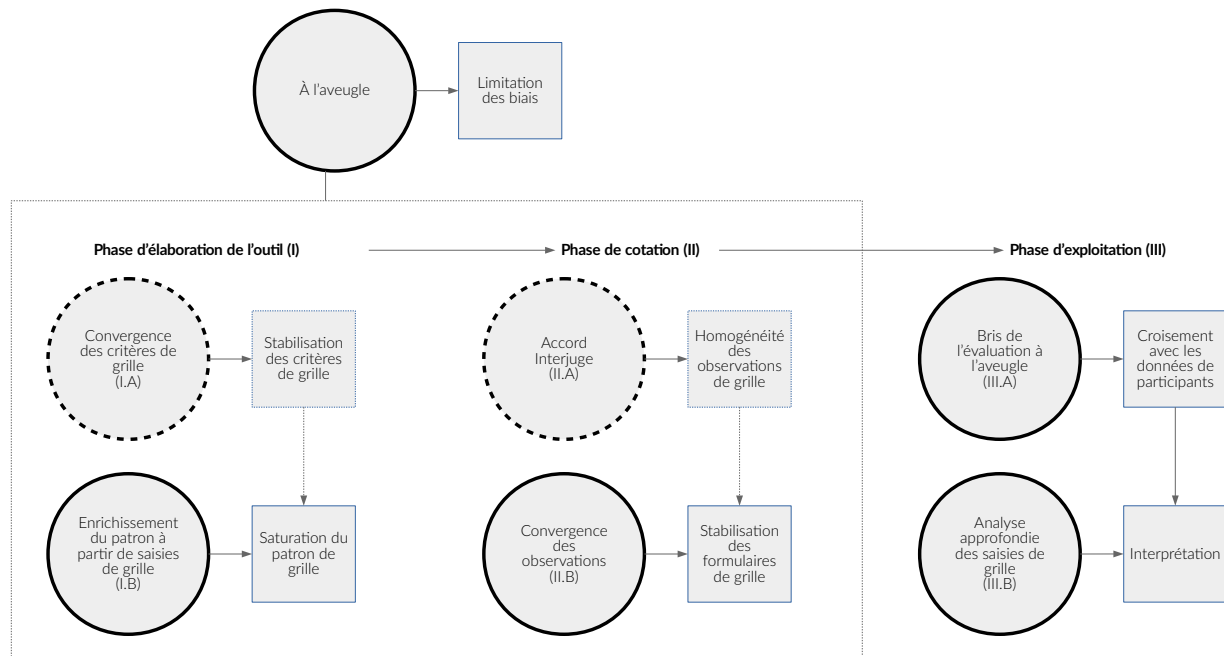


FIGURE 3.6 – Un cycle d'évaluation

l'autre, c'est pourquoi il n'a pas été jugé utile de l'inclure au corps de la description.

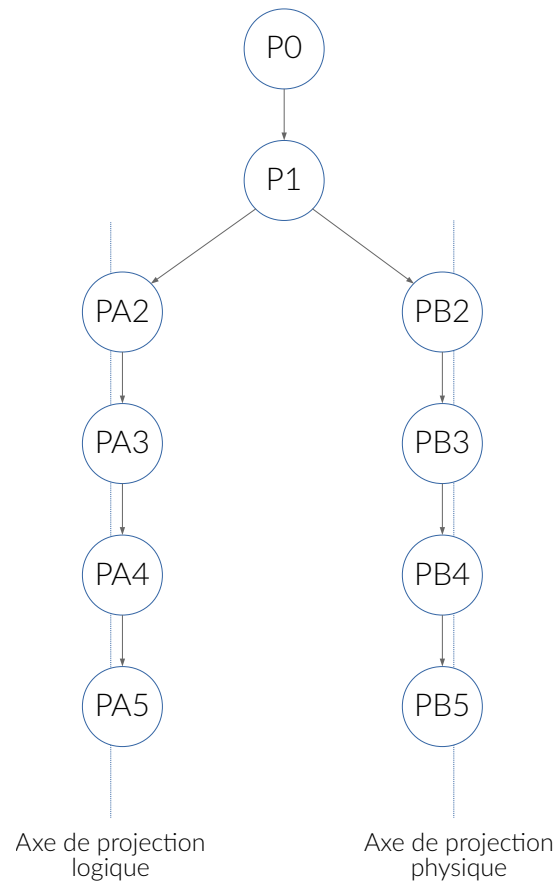


FIGURE 3.7 – Itérations des problèmes-archétypes

3.3 Plongée au cœur du problème

Dans cette section, nous approfondissons le problème par itérations successives, en traçant plusieurs sillons dans le **domaine du problème**. L'idée est de découvrir le périmètre du problème et d'envisager l'apparence extérieure d'éléments de la solution à mesure qu'on l'explore, plutôt qu'*a priori*. On se référera à l'**Annexe B** pour s'informer sur les notations utilisées ainsi qu'à l'**Annexe C** pour une description des *problem frames*.

La Figure 3.7 présente le schéma d'itération pour « plonger » dans le problème. Deux sillons parallèles (PA x et PB x) correspondent à deux projections distinctes du même problème : l'une sur un volet physique, qui met l'accent sur l'**environnement opératoire**, et l'autre sur l'aspect logique, c'est-à-dire sur le **domaine métier**. Par conséquent, PA5 et PB5 sont complémentaires, et projettent chacun deux aspects d'un même problème. Ils correspondent aux problèmes-archétypes finaux.

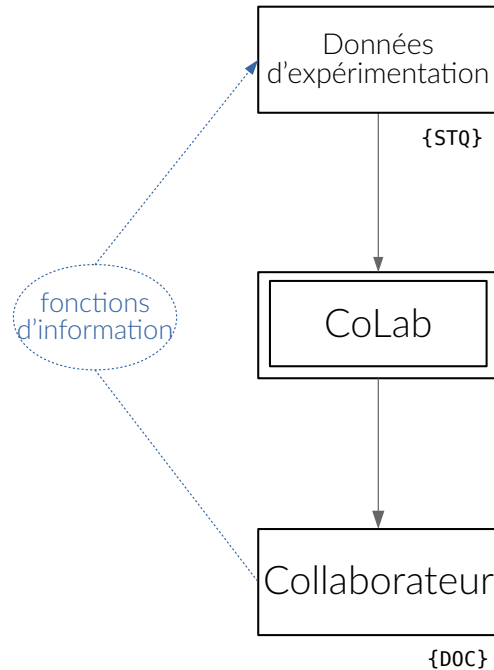


FIGURE 3.8 – Problème-archétype P0

3.3.1 P0, un simple système d'information

Problèmes. La description du domaine « nu » nous a amenés à identifier plusieurs problèmes. Tout d'abord, les vidéos enregistrées ne sont pas aisément transmises aux **collaborateurs** situés à Montréal². Ensuite, un déplacement des vidéos vers les **collaborateurs** augmente les risques de divulgation d'informations confidentielles. Il est à noter aussi que pour les analystes, la restructuration de grille est un processus fastidieux³. Enfin, l'acquisition vidéo n'est pas synchrone et le démarrage des sources d'acquisition (vidéo, audio et capteurs) se fait de manière séquentielle⁴.

Évolution On propose un « système d'information » (Figure 3.8). COLAB est alors un intermédiaire permettant à des **collaborateurs** du projet de consulter des données d'expérimentation⁵.

2. Une solution envisagée a été de mettre en place un serveur FTP sécurisé sous VPN, comme cela a été fait pour d'autres projets. Mais la procédure est lourde, et pas à la portée des moins « technodextres ».

3. C'est-à-dire coûteux en temps de travail. La principale raison est que d'une grille à l'autre, les annotations sont perdues.

4. Ce n'est pas optimal pour l'opérateur, et ça complexifie la reconstitution de la cohérence temporelle des données.

5. Dans le prototype 0.4, ces données sont des vidéos d'expérimentation ainsi que des informations horodatées sur les tâches effectuées par les participants. *Confer* section 4.1

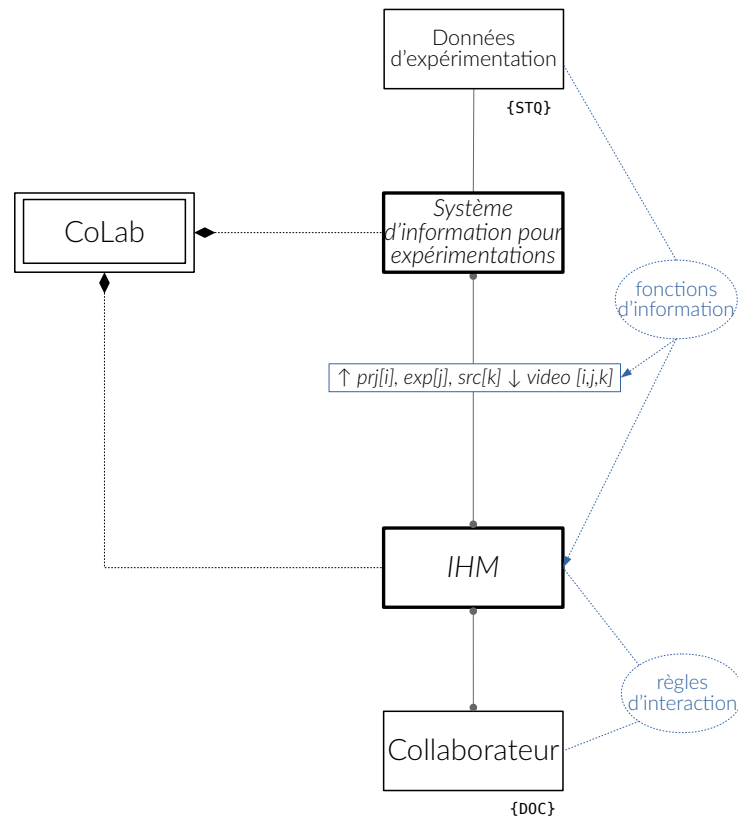


FIGURE 3.9 – Problème-archétype P1

Précisions. Dans cette version minimale, les données d'expérimentation se résument, pour une expérimentation i donnée, à un ensemble de capsules vidéo reliées unitairement à des sources de captation, c'est-à-dire les caméras disposées dans l'appartement. Ces données sont identifiées comme « statiques » car, dans une même expérimentation, elles ne varient plus une fois acquises. En nous plaçant dans un problème d'acquisition, nous aurions pu les qualifier d'« inertes » en considérant qu'une nouvelle expérimentation correspond à une mutation des données dans ce domaine.

3.3.2 P1, apparition de l'IHM

Problème. Il manque à P0 la possibilité pour l'utilisateur de sélectionner les données qu'il souhaite explorer.

Évolution. L'utilisateur peut choisir un projet $prj[i]$, une expérimentation $exp[j]$, une source de captation $src[k]$, puis consulter la vidéo $video[i, j, k]$ (Figure 3.9). Il peut à tout moment chan-

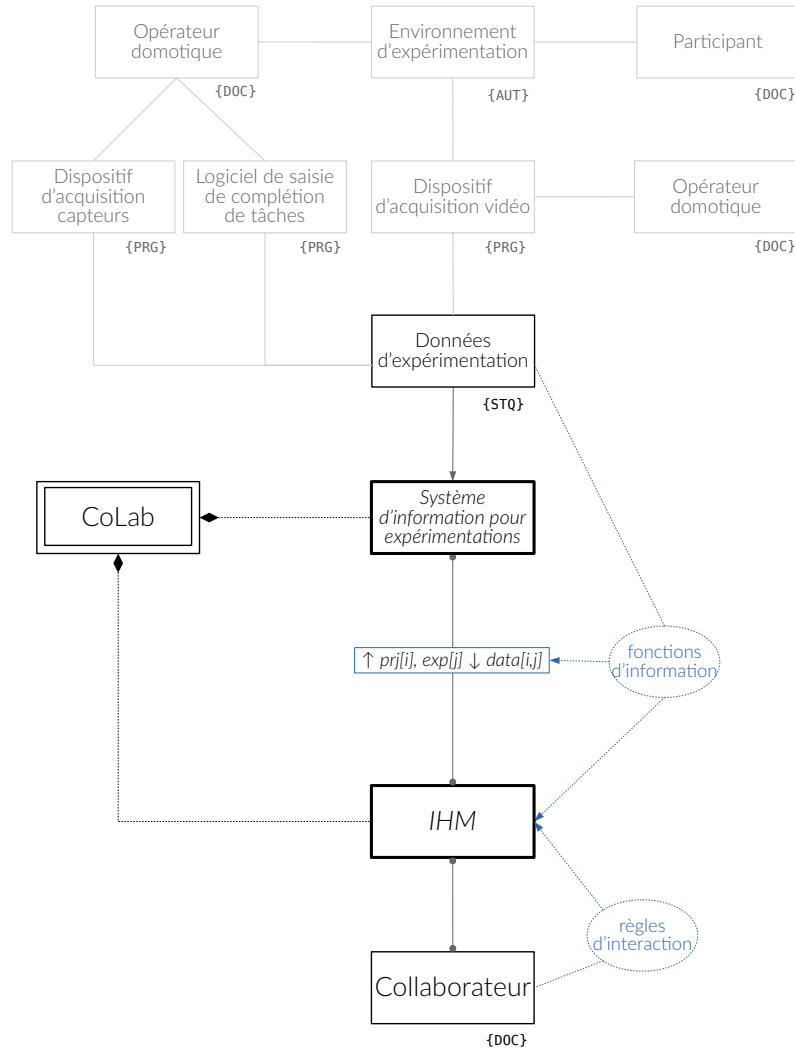


FIGURE 3.10 – Problème-archétype PB2

ger de source de captation en conservant le temps relatif depuis le début de l'expérimentation. On propose une version enrichie de l'archétype P0, en utilisant d'une part la variante « sur demande » du « système d'information », et en ajoutant d'autre part un système-connecteur centré (IHM, Figure 3.9).

3.3.3 PB2, élargissement du domaine du problème et projection physique

Problème. D'autres données doivent être exploitées par les **analystes comportementaux**. Notamment, les données relatives aux activités des participants, générées par l'**opérateur d'expéri-**

mentation ainsi que les données extraites des capteurs⁶.

Évolution. Nous creusons le domaine du problème en direction de l’environnement d’acquisition⁷ (Figure 3.10). On simplifie aussi le système d’information pour expérimentation en faisant apparaître une requête de sélection $prj[i], exp[j]$ qui collecte l’ensemble de données pertinentes $data[i, j]$ pour l’expérimentation j du projet i .

Précisions. En gris clair sont décrits des domaines qui se situent à l’extérieur du domaine du problème relatif à COLAB mais qui ne doivent pas être négligés pour autant, puisqu’ils peuvent en théorie influencer le domaine « Données d’expérimentation ». On pourra se référer à la sous-section 3.2.3 pour s’informer sur les activités qui y ont cours.

3.3.4 PB3, ajout d’une interface uniforme

Problème. Dans le problème PB2, nous avons décrit un environnement d’expérimentation et les différents outils et acteurs qui interagissent avec. Cette description est fortement couplée avec les dispositifs déployés au DOMUS, ce qui n’est pas satisfaisant compte tenu de l’objectif O.III (adaptabilité). Il est en effet obligatoire que COLAB puisse s’interfacer avec un large éventail d’environnements d’expérimentation.

Évolution. Nous devons introduire un nouveau système-solution qui résout le problème soulevé par l’objectif O.III (Figure 3.11). Ce système, appelé API SCÉNARII, remplace le système d’information et agit comme une interface générique entre COLAB et l’environnement d’acquisition. Le domaine et les exigences cernés de rouge ont un statut particulier. L’« Adaptateur d’environnement » est un système-transformationnel dépendant de l’API SCÉNARII mais dont l’implémentation doit être réalisée par l’équipe qui déploie l’API SCÉNARII. Ce système transforme des données locales d’expérimentation en des données canonisées.

3.3.5 PB4, essentialisation de l’environnement d’acquisition

Évolution. Nous allons donc séparer ces deux problèmes, l’adaptabilité d’une part, COLAB d’autre part (Figure 3.12). Pour réduire la portée du présent travail, nous considérons qu’un « modèle scénarii » contraint la structure des données transformées par API SCÉNARII, et c’est

6. On se référera à l’exploitation du questionnaire d’utilisabilité pour comprendre les motivations relatives à l’exploitation de ces données (section 4.2).

7. Nous faisons, en quelque sorte, une projection dans le domaine physique (environnement opératoire).

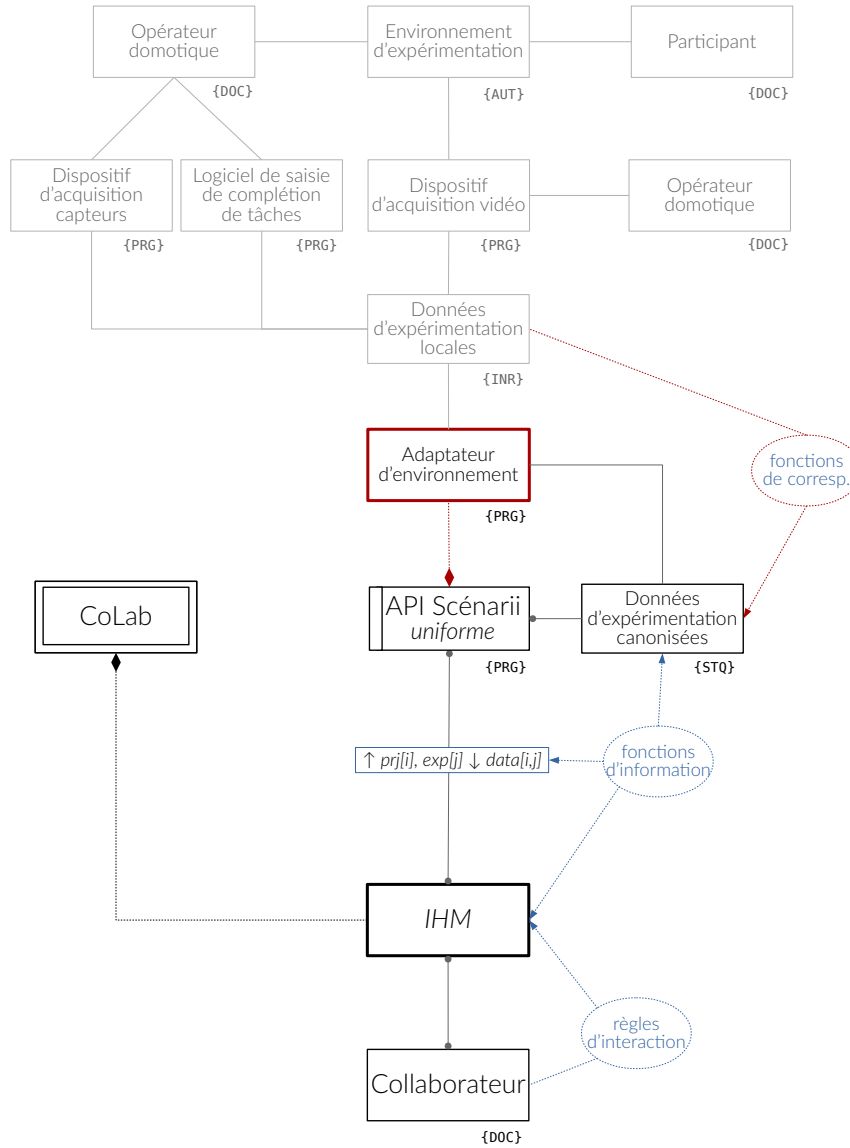


FIGURE 3.11 – Problème-archétype PB3

l'explicitation de ce modèle et des règles qui lui sont propres qui permettra à la couche d'adaptation visible en PB3 d'être réalisée. Nous faisons donc abstraction de l'environnement d'expérimentation, qui se retrouve essentialisé dans ces règles.

Précisions. Le domaine « Données d'expérimentation canonisées » est inaltérable (statique), c'est-à-dire que des données préalablement générées ne peuvent pas être modifiées par le système d'information. Du point de vue de COLAB et de n'importe quel autre client de l'API SCÉNARII, c'est un domaine en lecture seule. On se référera à la [section 3.6](#) pour prendre connais-

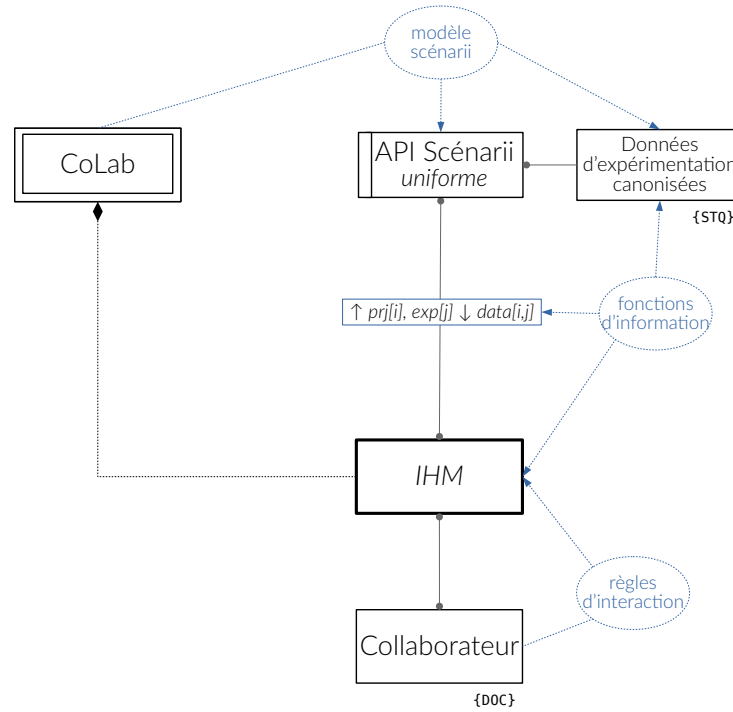


FIGURE 3.12 – Problème-archétype PB4

sance du modèle scénarii.

3.3.6 PB5, ajout d'un modèle organisationnel

Problème. PB4 nous a permis de séparer deux problèmes distincts. Mais il nous manque à définir l'articulation entre les deux systèmes-solution. En effet, nous n'avons pas de représentation physique de la répartition spatiale des éléments du système-solution et des domaines périphériques.

Évolution. Pour se conformer à la caractéristique d'accessibilité (C.I) et pour satisfaire l'objectif de gouvernance (O.IV), on propose de déployer COLAB dans une structure centralisée (Figure 3.13). Chaque organisme participant $org(n)$ détient sa propre API SCÉNARII(n). Un système d'information — le « système d'information central » — vient s'intercaler entre le client et les multiples APIs SCÉNARII. Son rôle est d'acheminer les requêtes vers la bonne API SCÉNARII. Enfin, le modèle organisationnel décrit les relations entre les différents éléments constitutifs de l'architecture déployée.

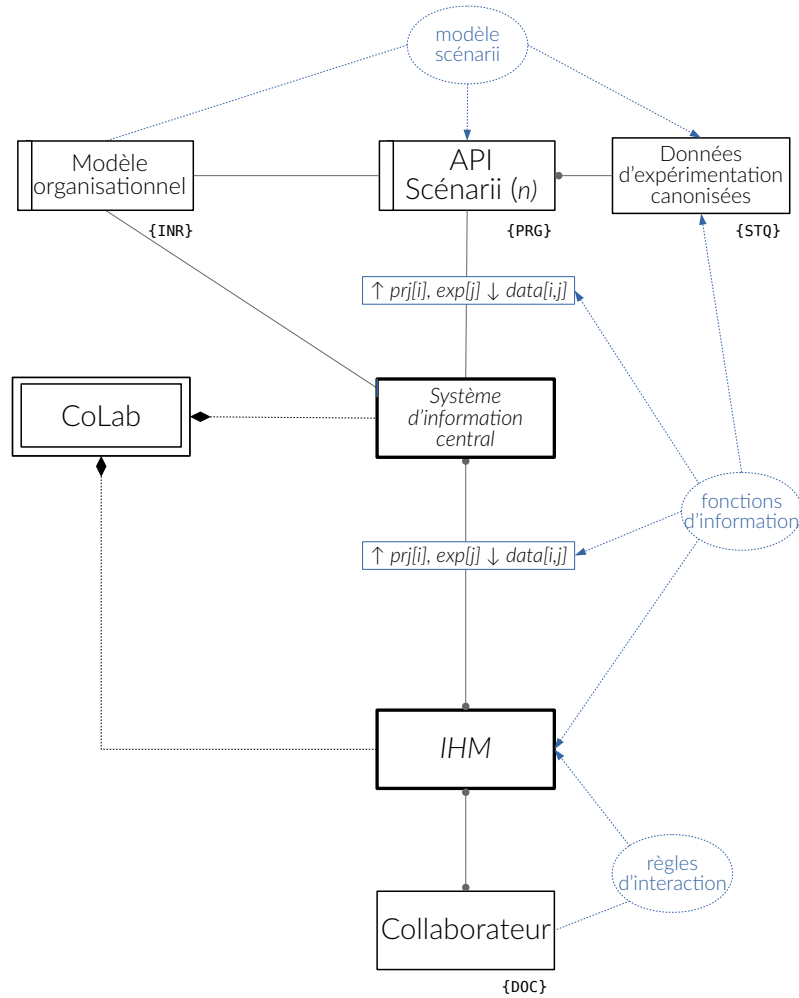


FIGURE 3.13 – Problème-archétype PB5

Précisions. On se référera à la section 3.4 pour s'informer sur l'architecture physique et le modèle logique organisationnel.

3.3.7 PA2, obligation d'un contrôle d'accès

Problème. Il manque à P1 et P2 des éléments de sécurité pour satisfaire la caractéristique fondamentale C.II (confidentialité). Il est en effet nécessaire :

- d'authentifier un utilisateur dans le système ;
- de contrôler le périmètre d'action d'un utilisateur dans le système.

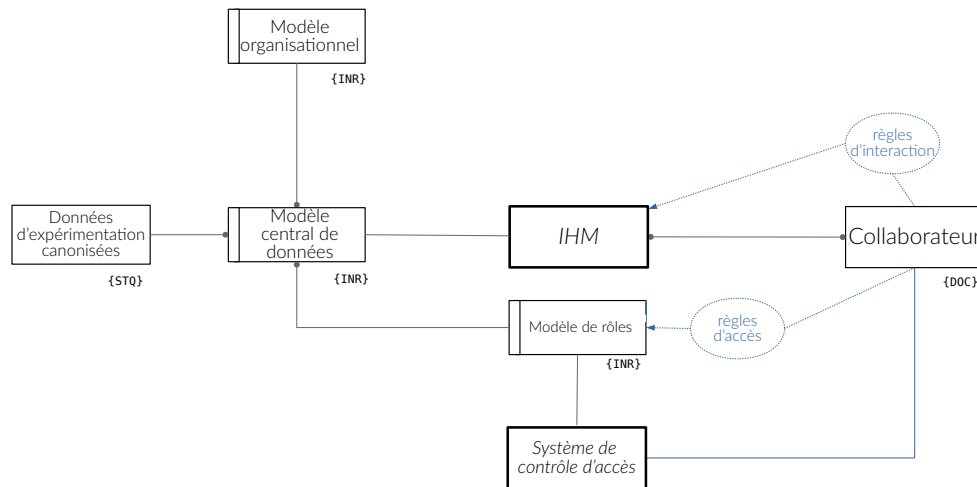


FIGURE 3.14 – Problème-archétype PA2

Évolution. Un système-contrôleur, le « système de contrôle d'accès », est ajouté à COLAB (Figure 3.14). Il restreint l'accès à certaines ressources et réduit le champ d'action de l'utilisateur, suivant les prérogatives décrites dans le domaine « modèle de rôles ». Par exemple, un utilisateur doit être **collaborateur** d'un **projet** pour accéder aux vidéos d'expérimentations relatives à ce **projet**.

Précisions. On se référera à la sous-section 3.4.1 pour prendre connaissance du modèle de rôles. Aussi, pour des raisons de lisibilité, on a supprimé le système-solution « COLAB » du diagramme, en ne laissant apparaître que ses composants, identifiés par les rectangles à bordure noire épaisse.

3.3.8 PA3, intégration de l'analyse comportementale

Problème. Il manque à PA2 le domaine décrivant l'Analyse Comportementale (AC). En effet, les **analystes** doivent pouvoir définir, utiliser et exploiter les outils pour l'AC.

Évolution. C'est « l'atelier d'AC » qui permet de créer et manipuler ces outils (Figure 3.15). La Figure 3.16 est une projection de ce système-atelier en trois micro ateliers. Le premier, l'« assistant à la création d'Outil d'Analyse Comportementale (OAC) », permet de définir :

- des patrons de grille ;
- des catégories d'observation ;
- des Modèles de Tâches pour Analyse Comportementale (MTACs).

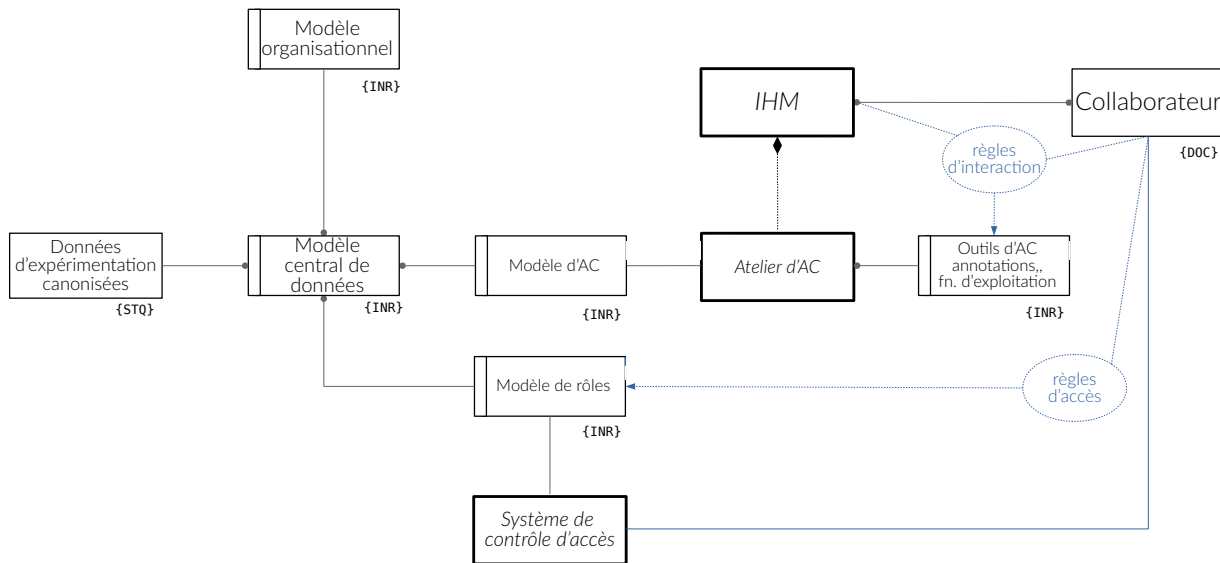


FIGURE 3.15 – Problème-archétype PA3

Ces trois items — des « outils d’AC » — sont des structures d’arbre définies par l’*analyste comportemental*. Ensuite, l’« atelier d’AC pour expérimentation » permet de faire usage des outils d’AC⁸. Dans cet atelier, l’*analyste comportemental* peut, pour une *expérimentation* en particulier, saisir diverses *annotations* :

- des *observations de verbatim* ;
- des *tâches réalisées* ;
- un *formulaire de grille* et ses critères, comme des *observations de grille*.

Enfin, l’« éditeur de fonctions d’exploitation pour AC » sert à définir des fonctions d’exploitation statistiques et documentaires à partir des *annotations*.

Précisions. Le détail de ces structures d’arbres est présenté dans la sous-section 3.5.1.

3.3.9 PA4, exploitation des données d’analyse comportementale

Problème. Une fois les *annotations* saisies, elles doivent pouvoir être exploitées par les *collaborateurs*.

Évolution. On envisage donc un premier système transformationnel, le « module d’exploitation », chargé d’agréger ces données pour en tirer des statistiques et des rapports synthétiques.

8. C’est pourquoi une flèche pointillée — exprimant la contrainte — va du domaine « Outils pour AC » (OAC) au domaine « annotations » (Figure 3.16)

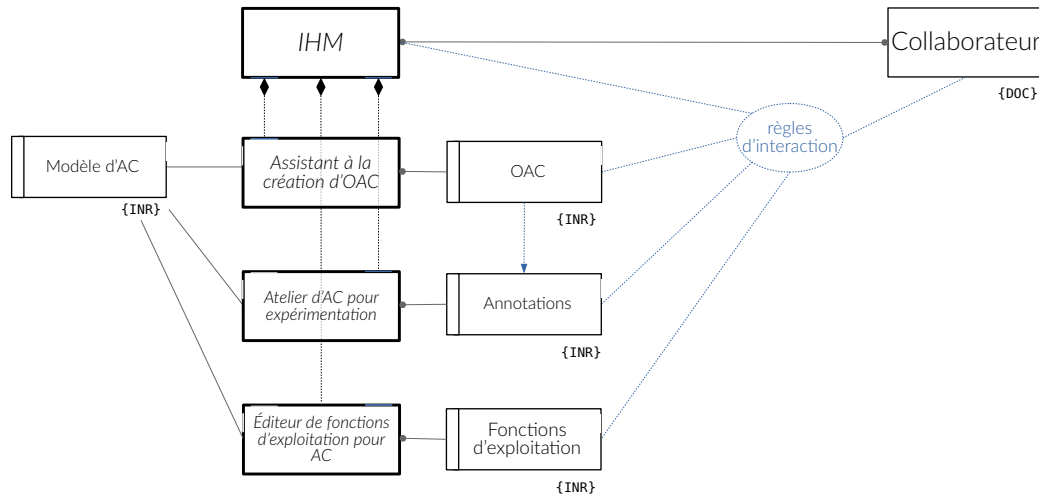


FIGURE 3.16 – Projection de l'atelier d'AC sur les trois étapes d'un cycle d'évaluation

Puis, un deuxième module transformationnel, le « module d'export », permet de générer des fichiers portables utilisables par d'autres logiciels (par ex., xslt, docx, pdf) (Figure 3.17).

La Figure 3.18 détaille l'interfaçage entre ces deux systèmes transformationnels. On remarquera la présence de l'atelier « éditeur de fonctions d'exploitation pour AC » et du domaine

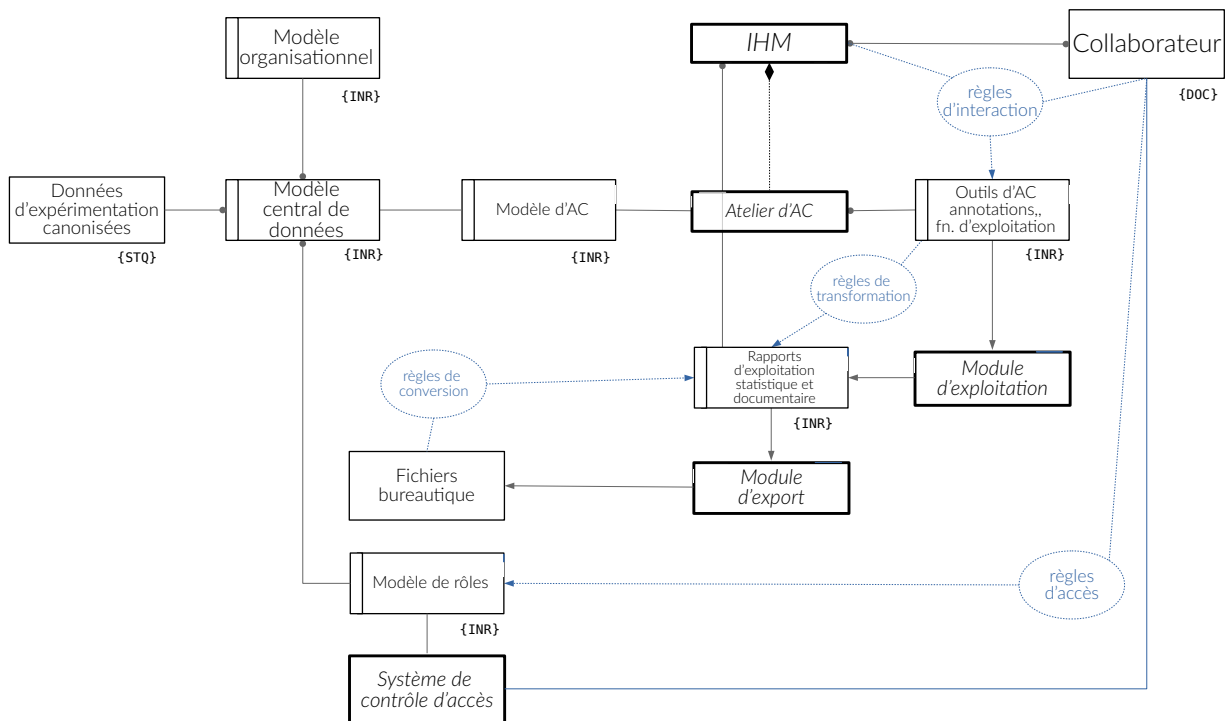


FIGURE 3.17 – Problème-archétype PA4

qui lui est associé, « fonctions d'exploitation », issus de la [Figure 3.16](#). Ces derniers, créés par un *collaborateur*, contraignent en effet la structure des rapports générés (« rapports d'exploitation statistique et documentaire »), puisqu'ils définissent les fonctions de transformations des données d'annotation.

3.3.10 PA5, interfaces de gestion

Problème. COLAB nécessite une certaine coordination entre ses membres, d'où le besoin de configurer et d'administrer quatre niveaux de coordination :

- inter organismes (global) ;
- organisme ;
- par projet ;
- par étude.

Évolution. On ajoute donc des « Ateliers de configuration » ([Figure 3.19](#)). On peut projeter ces derniers sur les quatre niveaux de coordination ([Figure 3.20](#)).

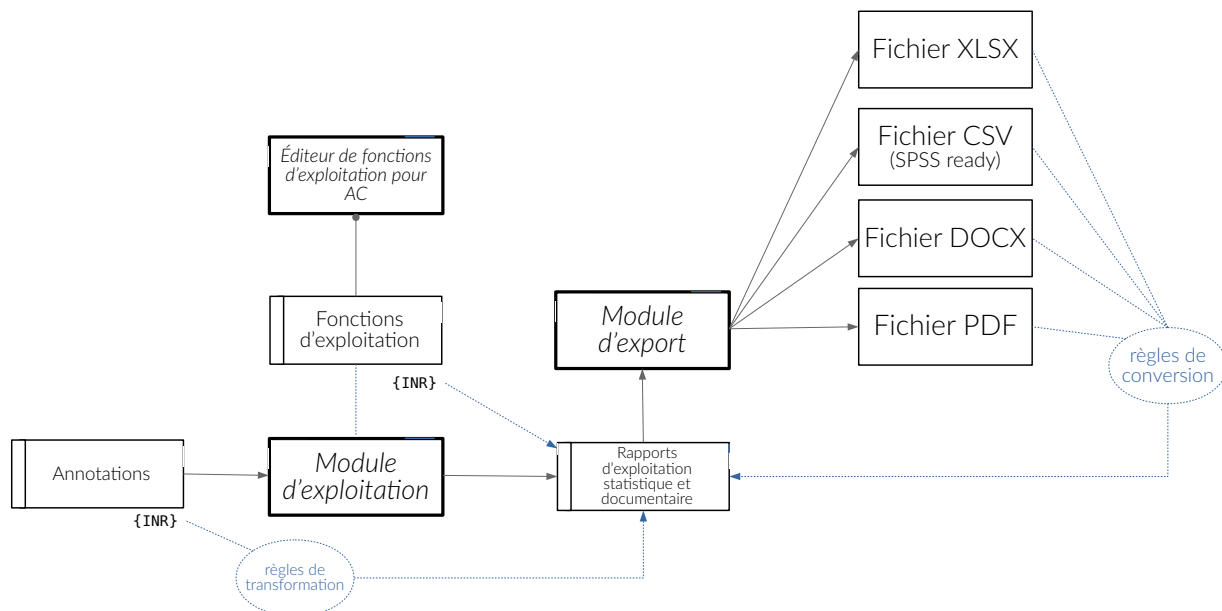


FIGURE 3.18 – Projection des systèmes transformationnels « module d'exploitation » et « module d'export »

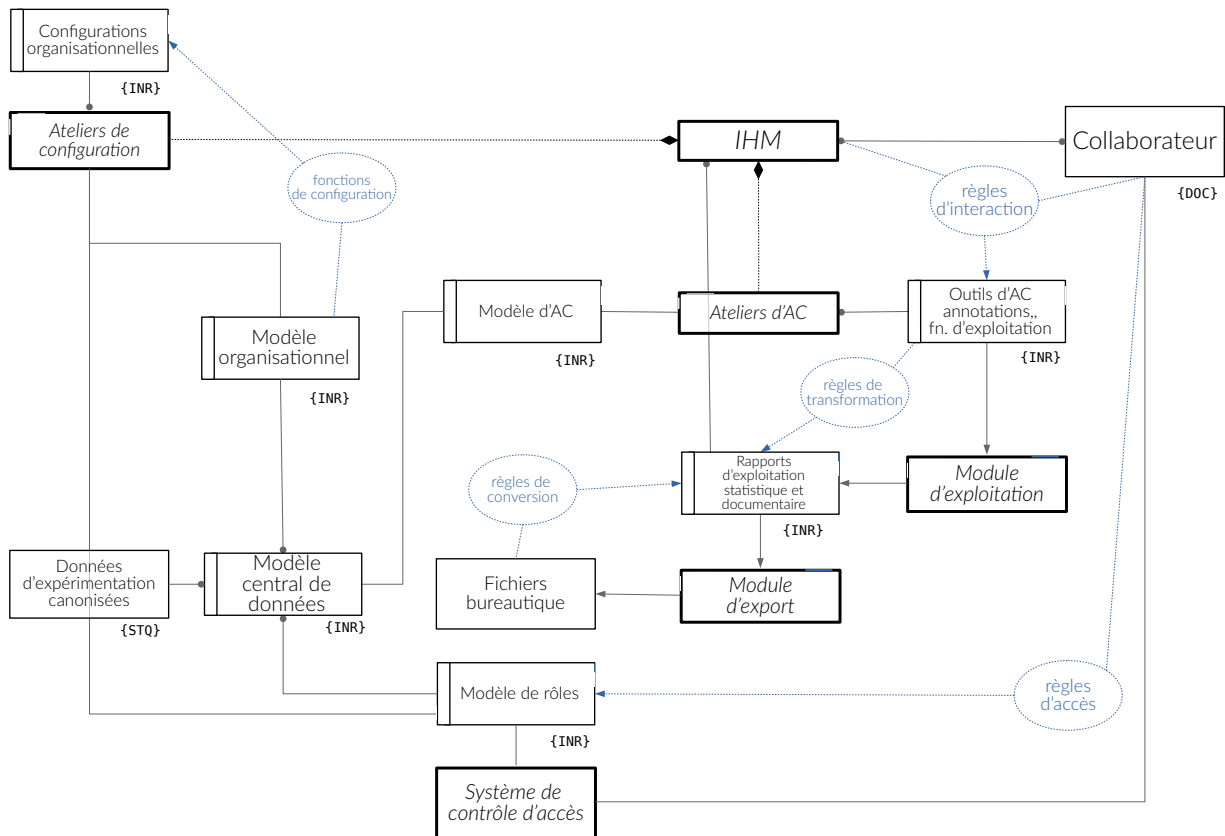


FIGURE 3.19 – Problème-archétype PA5

Précisions. Dans la Figure 3.20, on a enlevé les exigences pour améliorer la lisibilité. Elles correspondent, pour chaque atelier, aux règles d'édition permises sur l'objet de configuration. On remarquera aussi que l'administration globale ne se fait pas via COLAB, mais via une **Interface en Ligne de Commande (ILC)** dédiée. On se référera à la section 3.4 pour se documenter sur le modèle de rôles.

3.3.11 Synthèse des composants découverts

Dans le Tableau 3.2, les différents composants physiques identifiés pour COLAB sont donnés. Les composants logiques sont présentés dans le Tableau 3.3.

TABLEAU 3.2 – Composants système physiques de CoLAB

Composant système physique	Code	Archétype
Système d'information central	SIC	Système d'information
IHM	IHM	Connecteur
API SCÉNARII <i>n</i>	APISc	Connecteur et système d'information
ILC d'administration globale	ILCAG	Atelier et système-contrôleur

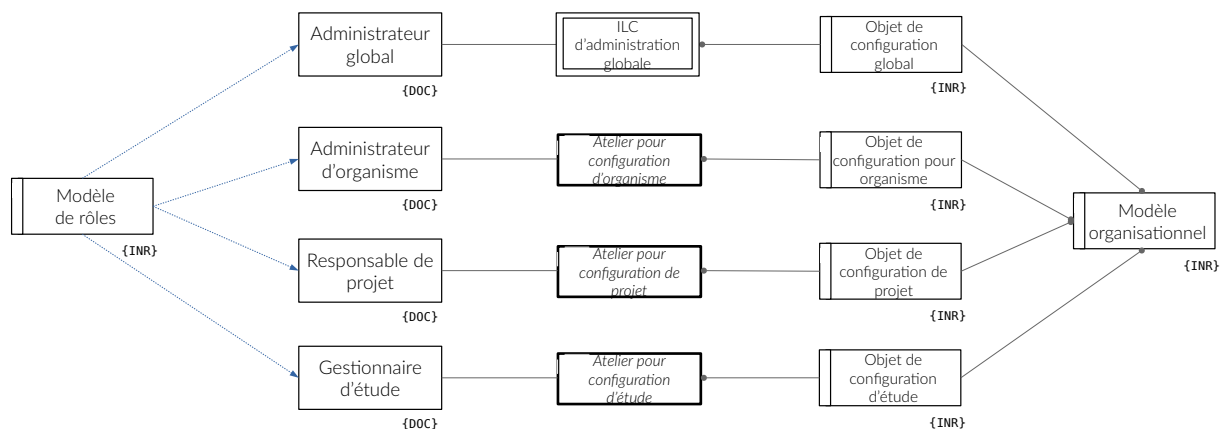


FIGURE 3.20 – Projection des « ateliers de configuration » sur les quatre niveaux de coordination

TABLEAU 3.3 – Composants système logiques de COLAB

Composants système logiques	Code	Archétype
Système de contrôle d'accès	LO1	Système-contrôleur
Atelier d'analyse comportementale	LO2	
→ Assistant à la création d'OAC	LO2A	Atelier
→ Atelier d'AC pour expérimentation	LO2B	Atelier et système d'information
→ Éditeur de fonctions d'exploitation pour AC	LO2C	Atelier
Module d'exploitation	LO3	Système transformationnel
Module d'export	LO4	Système transformationnel
Atelier de configuration	LO5	
→ Atelier de configuration d'organisme	LO5A	Atelier, système d'information et système-contrôleur
→ Atelier de configuration pour projet	LO5B	Atelier, système d'information et système-contrôleur
→ Atelier de configuration pour étude	LO5C	Atelier et système-contrôleur

3.4 Modélisation organisationnelle et sociale

3.4.1 Modèle de rôles

TABLEAU 3.4 – Définition des responsabilités dans COLAB

Responsabilité	Description
administrateur global	<i>CoLab:Responsabilité.</i> Un administrateur possède un contrôle total sur l'application, à l'exception des études. Il est notamment en charge d'autoriser la configuration de nouveaux organismes.
administrateur d'organisme	<i>CoLab:Responsabilité.</i> Un administrateur d'organisme est en charge de la configuration de l'API SCÉNARII. Il choisit aussi la politique d'inscription des projets sous la responsabilité de l'organisme. Il a le pouvoir de restreindre l'accès et supprimer des données d'expérimentation relatives à l'API SCÉNARII dont il est responsable.
opérateur en domotique	<i>CoLab:Responsabilité, API Scénarii:Responsabilité.</i> Un opérateur domotique est en charge du bon fonctionnement des dispositifs d'acquisition du laboratoire où ont lieu les expérimentations (le site). Il valide l'expérimentation d'un point de vue technique, et documente des erreurs d'acquisition si nécessaire.
responsable de projet	<i>CoLab:Responsabilité.</i> Un responsable de projet a la charge de la gestion des collaborateurs du projet.
collaborateur	<i>CoLab:Responsabilité.</i> Un collaborateur est un utilisateur qui a été intégré à un projet. Il dispose alors de droits d'accès aux données d'expérimentation pour les projets dans lesquels il est collaborateur. Il peut aussi y créer des études.
gestionnaire d'étude	<i>CoLab:Responsabilité.</i> Il est en charge d'une étude. Il doit notamment choisir les outils d'analyse utilisés par le groupe d'étude, fixer des jalons et assurer la coordination entre les associés.
associé d'étude	<i>CoLab:Responsabilité.</i> Un associé d'étude est un utilisateur qui a été intégré dans une étude. Il peut notamment produire des annotations à partir des OAC déployés pour l'étude.

— suite à la page suivante —

Définition des responsabilités dans COLAB – suite

Responsabilité	Description
visiteur d'étude	<i>CoLab:Responsabilité</i> . Un visiteur d'étude est un utilisateur qui a des permissions dégradées par rapport à l'associé d'étude. Il peut consulter les expérimentations, les annotations et les résultats, mais en lecture seule, sans possibilité d'édition. Il a cependant le droit de participer aux discussions de l'espace prévu à cet effet.

Le contrôle d'accès choisi est l'*Object Scope Delegate Role Based Access Control* (OSD-RBAC). On se référera à la sous-section 4.4.4 pour plus d'informations sur ce type de contrôle d'accès. La **Tableau 3.4** définit chaque rôle (responsabilité) tandis que la **Figure 3.21** propose une représentation hiérarchique de la structure de rôles. Dans cette dernière, on peut observer les classes déléguées (Organisme, Projet et Étude).

Le **Tableau 3.5** présente une vue partielle de la délégation dans COLAB. On peut y observer que les droits sur *annotations* peuvent être délégués à un *utilisateur* ou à une *étude*. C'est le *gestionnaire d'étude* qui choisit le modèle de propriété : collectif (Étude) ou individuel (Utilisateur). Quant à la classe Étude, la délégation est partagée entre la classe Utilisateur pour la modification, et la classe Projet pour la création.

TABLEAU 3.5 – Délégation de portée dans COLAB

Classe paresseuse	Classe déléguée
Utilisateur	Utilisateur
Projet	Projet
Expérimentation	Projet
Post	Utilisateur
Commentaire	Utilisateur
Annotation	Étude \vee Utilisateur
Étude	Utilisateur (modification) \wedge Projet (création)

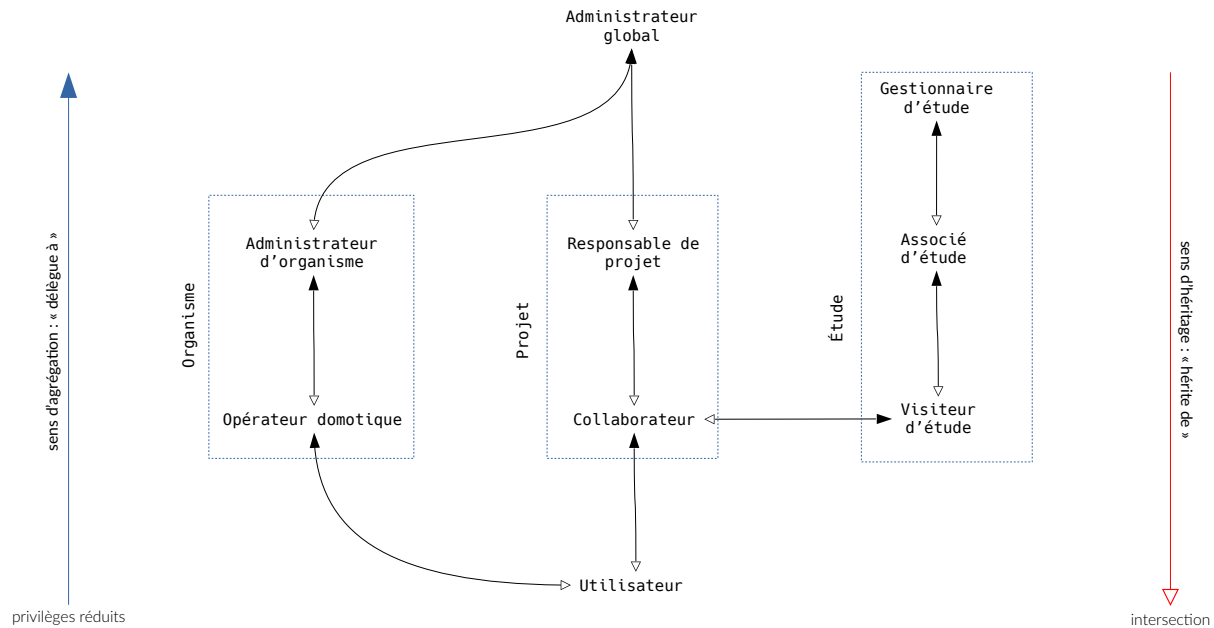


FIGURE 3.21 – Hiérarchie de rôles dans CoLab avec délégation de classes

3.4.2 Modèle coopératif

La Figure 3.22 décrit la distribution physique des **organismes** et des **collaborateurs** par rapport à la coopérative académique en charge de COLAB. Comme on peut l'observer, les **collaborateurs** sont affiliés à des **organismes**, à l'exception du collaborateur 3, non-affilié. Chaque organisme a sa propre API SCÉNARIII interfacée avec le serveur central administré par la coopérative. On se référera à la section 5.2 pour découvrir en détail le modèle de gouvernance proposé.

3.4.3 Modèle organisationnel

La Figure 3.23 présente les différentes entités qui structurent le modèle organisationnel propre à COLAB. Pour le parcourir, on peut partir de l'**organisme** en charge d'une API SCÉNARIII. L'**organisme** a la responsabilité de gérer un **projet**. Ce dernier est composé d'un ensemble d'**expérimentations**, qui peuvent être réparties sur différents sites. Dans un **projet**, une **étude** peut être créée à laquelle sont associés des **OAC**. On remarquera que ce modèle permet les projets avec **expérimentations** multisites.

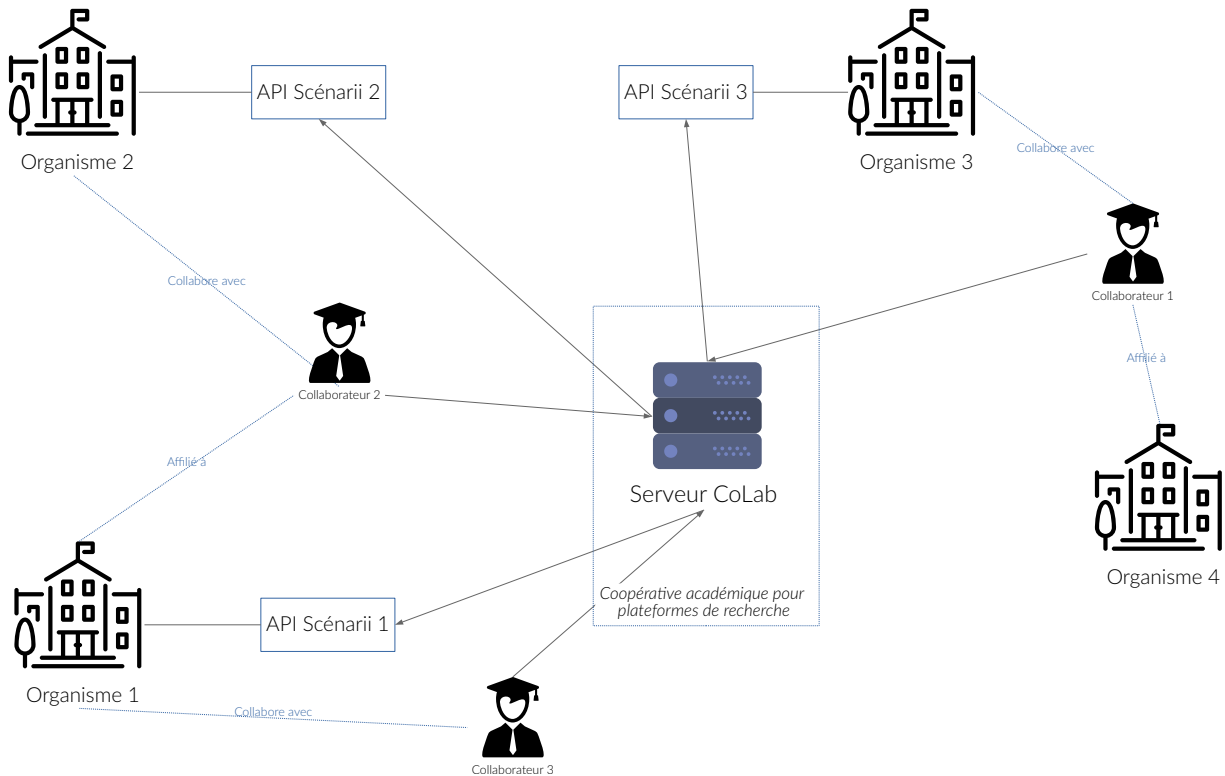


FIGURE 3.22 – Modèle du coopérativisme académique

3.4.4 Modèle de collaboration

Tandis que le **projet** encadre les accès aux données d'expérimentation, l'**étude** est le lieu privilégié de la collaboration. Dans une telle étude, le **gestionnaire** peut déterminer les **outils** à mettre en place. Ces derniers peuvent être issus du **magasin libre des universités**. Dans ce cas de figure, le **gestionnaire d'étude** a le choix entre le *forker*⁹ pour devenir éditable par tous les **associés d'étude** ou bien le conserver tel quel, auquel cas il n'est pas éditable par les **associés d'étude**. S'il est *forké*, l'outil appartient au groupe d'**étude** et non plus à un propriétaire nominal. Lorsqu'il est stabilisé et de nouveau figé, il peut être publié.

Au début d'une **étude**, le **floutage d'expérimentation** est activé. Ce dernier est une condition *sine qua non* pour que l'évaluation soit faite à l'aveugle, et seul le **gestionnaire d'étude** peut choisir de lever le floutage. À la fin d'une **étude**, les résultats sous forme de rapports statistiques peuvent être publiés, c'est-à-dire accessibles sans authentification et partagés sous forme d'URLs uniques, auquel cas des chercheurs intéressés pour réviser l'étude peuvent demander

9. Dans les logiciels de versionnement, un *fork* désigne un embranchement, une bifurcation dans l'historique du code source. Par extension, on parle de *fork* pour désigner un logiciel issu d'un autre, développé en toute indépendance du dernier. Par exemple, *LibreOffice* est un *fork* de *OpenOffice*.

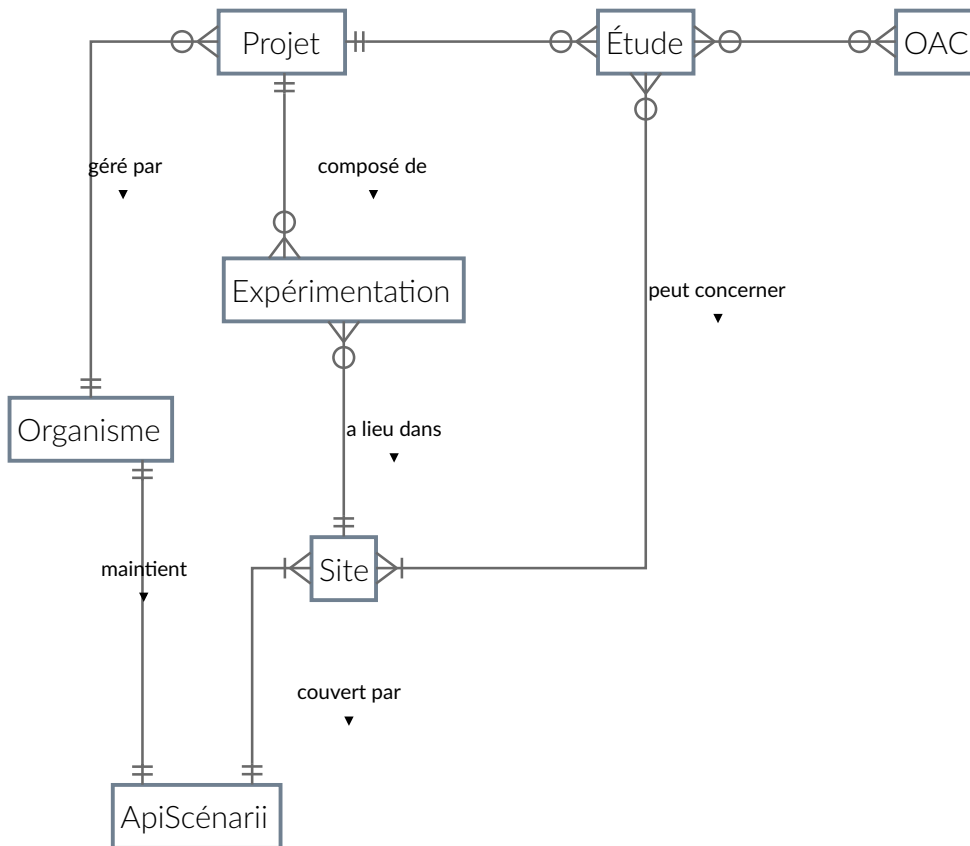


FIGURE 3.23 – Modèle logique organisationnel

à devenir **visiteur d'étude**. Aussi, le **gestionnaire d'étude** peut activer ou désactiver les **évaluations réflexives** pour un **OAC** déployé, auquel cas les **annotations** qui obtiennent de faibles notes doivent attirer l'attention sur des difficultés à appliquer les outils d'analyse dans certaines situations.

Le **gestionnaire d'étude** peut aussi créer une feuille de route. Dans cette dernière, des jalons sont fixés. Ces jalons concernent une partie ou l'intégralité des **expérimentations** à évaluer. Une feuille de route pourrait ressembler à cette séquence de jalons :

- J1** rédiger le verbatim de toutes les **expérimentations** ;
- J2** stabiliser un **patron de grille** sur dix **expérimentations** ;
- J3** **cotation de grille** sur toutes les **expérimentations** avec accord interjuge en binômes ;
- levée du **floutage d'expérimentation** pour analyse.

Le dernier élément n'est pas un jalon, c'est une transition automatique lorsque le jalon 3 est atteint. Ces jalons offrent la possibilité au **gestionnaire d'étude** d'auto assigner les tâches à réaliser

par les **associés d'étude**. Sinon, le modèle de collaboration prend la forme d'« une pioche » de tâches à effectuer, garantissant que deux **associés d'étude** ne puissent pas travailler sur la même **expérimentation** en même temps.

En plus de ces éléments, des fonctionnalités sociales sont présentes. Un espace de discussion rattaché à l'**étude** est proposé, prenant la forme d'un forum moderne avec notifications. Un espace similaire existe pour le **projet**, dans lequel les **administrateurs d'organisme**, **opérateurs en domotique** et bien entendu les **collaborateurs** peuvent participer. Ce dernier a plutôt pour vocation la comparaison entre les groupes d'étude et les questions relatives à l'environnement d'acquisition. Dans ces espaces, les **utilisateurs** peuvent facilement se référer à une **expérimentation** ainsi qu'à un moment d'expérimentation, avec par exemple #ANONYMOUS18@13m18s. Les **annotations** aussi peuvent être commentées par les différents **associés d'étude**, afin d'améliorer la surface d'échanges possibles et la collaboration. Les **OAC** ont eux aussi un espace de commentaires dédié à la critique et l'amélioration de l'outil. De plus, le modèle de propriété pour **annotations** peut être choisi par le **gestionnaire d'étude**. Soit les annotations appartiennent à leurs auteurs, auquel cas elles ne peuvent pas être éditées par d'autres **associés d'étude**, soit elles appartiennent au groupe d'**étude**, auquel cas elles peuvent être modifiées par n'importe quel **associé d'étude**. Dans cette dernière situation, l'auteur initial est tout de même notifié de la modification. Enfin, une phase d'évaluation délimitée par un jalon de **cotation de grille** peut être définie avec **accord interjuge** en binôme. Dans ce cas, chaque **expérimentation** doit être annotée par les deux **associés** d'une même paire. Les **observations de grille** atomiques rédigées par un binôme sont visibles par l'autre, mais ni la **catégorie**, ni l'éventuelle description associée ne sont visibles. À la fin de cette phase, une synthèse des principales différences est donnée, avec un score de fidélité. Les différences sont résolues à l'unanimité.

Enfin, des notifications sont accessibles à chaque **utilisateur**, ainsi qu'un tuple d'actions possibles pour chacune (**Tableau 3.6**). Parmi celles-ci, l'action « reporter » permet de convertir une notification en « tâche » à effectuer par l'**utilisateur**, accessible depuis son tableau de bord.

TABLEAU 3.6 – Différents types de notifications

Type	Description	Actions possibles
notifications sociales	Elles sont relatives à des échanges sur les espaces de discussion.	Voir la discussion, Marquer comme lu, Reporter
notifications de coopération	Elles sont relatives à des éditions collaboratives (<i>annotations</i> , <i>OACs</i>).	Voir la modification, Commenter, Marquer comme lu, Ne plus m'avertir
notifications de responsabilité	Elles sont relatives à des demandes de permission en groupe.	Accepter, Refuser, Reporter

3.5 Modélisation de l'analyse comportementale

3.5.1 Proposition d'un modèle pour grilles d'évaluation

La *grille d'évaluation* préexiste sous une forme « papier » assez libre, et on propose ici de la définir d'une manière plus contraignante pour permettre ensuite l'implémentation d'un *modèle analogique*. La Figure 3.24 décrit un *patron de grille*, c'est-à-dire son squelette, inspiré d'un travail réalisé par des étudiantes de l'Université de Montréal. On se propose dans le Tableau 3.7 de définir toutes les composantes de la grille¹⁰.

Cette dernière constitue « un outil générique qui permet aux *analystes comportementaux* d'évaluer le comportement des *participants* à partir de critères objectifs qu'ils ont pu définir ». Elle se décompose en deux types d'items :

- le *patron de grille*, qui normalise l'évaluation ;
- un *formulaire de grille*, qui est dérivé du *patron de grille*. Il correspond aux saisies de *critères de grille* pour une *expérimentation* particulière.

On distingue ensuite deux types de *critère de grille* :

- les critères observables. Ces critères nécessitent l'intervention d'un *analyste comportemental* pour être saisis ;
- les critères inférés. Ces critères sont automatiquement calculés par des fonctions d'autres critères.

Par exemple, dans la Figure 3.24, le critère inféré « Temps d'exécution pour expérimentations » est calculé comme la somme des temps d'exécution de chaque tâche : $t_{exec} =$

10. Une partie de ces concepts sont créés *ad hoc* pour construire le modèle.

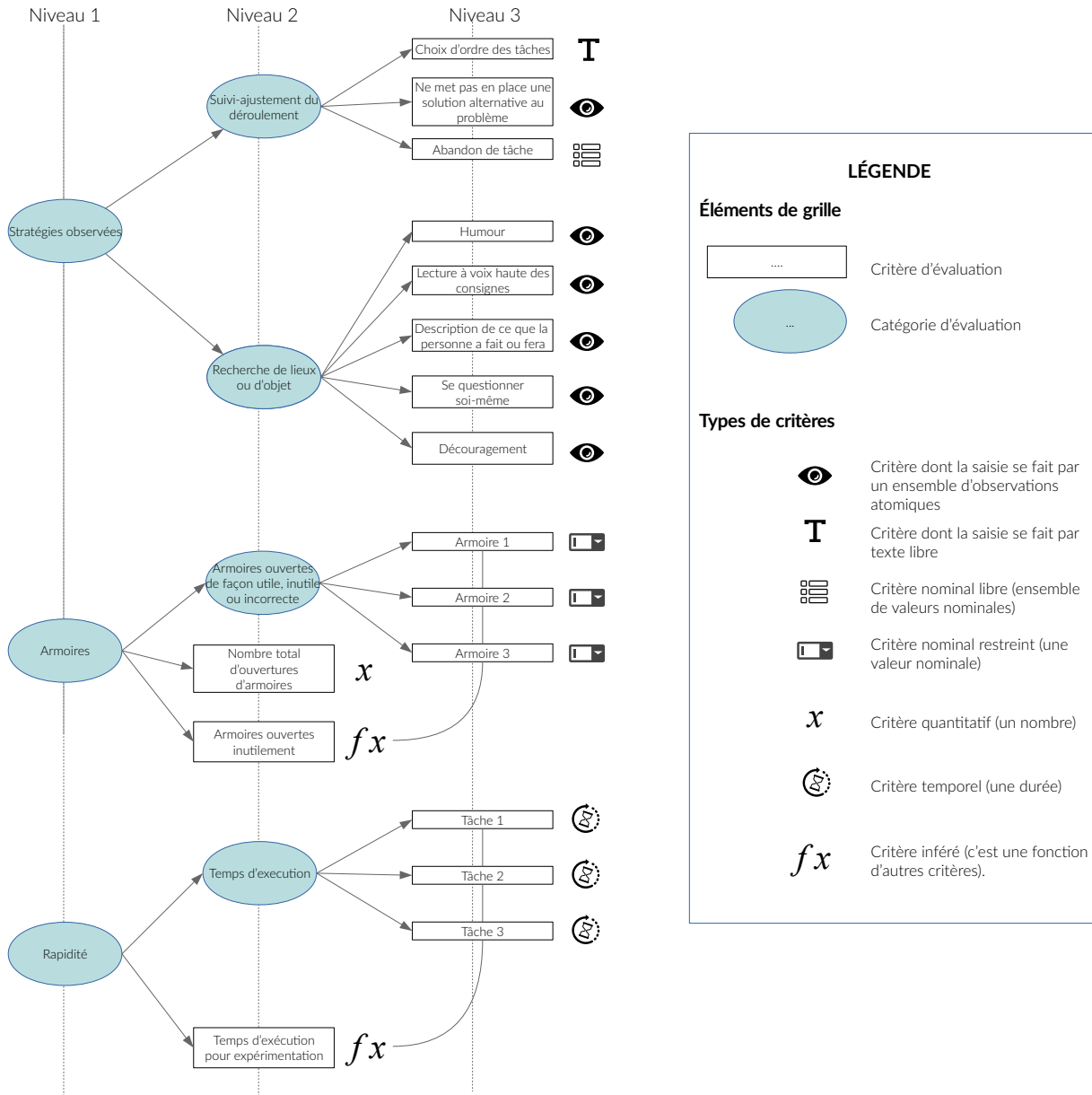


FIGURE 3.24 – Structure d'un patron de grille

$\sum_{tache=1}^n temps_execution(i)$. Un critère inféré peut être comparé à une expression de calcul dans un tableau. À chaque critère est associé un type :

- ensemble d'observations atomiques. À chaque observation est attribué un temps relatif depuis le début d'expérimentation. Ce sont ces observations qui seront visibles dans l'explorateur temporel ;

TABLEAU 3.7 – Terminologie détaillée de la grille d'évaluation

Terme	Description
patron de grille	<i>CoLab</i> . Ensemble de critères structurés sous forme d'arbre dont les feuilles sont des critères de grille, et dont les nœuds de niveaux inférieurs constituent des catégories d'autant plus générales qu'elles sont proches de la racine. Les nœuds servent donc à structurer et abstraire la grille tandis que les feuilles permettent de dériver le formulaire de grille.
critère de grille	<i>CoLab</i> . Un critère de grille est une feuille du patron de grille qui va correspondre, pour une expérimentation donnée, à une saisie par l'évaluateur. Un critère de grille est toujours typé, c'est-à-dire qu'il contraint les valeurs possibles de la saisie (par ex., texte, valeurs modales).
cotation de grille	<i>Ergothérapie, CoLab</i> . Application du patron de grille à une expérimentation particulière par l'intermédiaire du formulaire de grille. Une analogie peut être faite avec un questionnaire, qui serait le patron, et les réponses au questionnaire, qui seraient les formulaires de grille.
formulaire de grille	<i>CoLab</i> . Le formulaire de grille est dérivé du patron de grille et permet de saisir des critères de grille pour une expérimentation en particulier. L'activité qui consiste à saisir ce formulaire est appelée cotation de grille.
observation de grille	<i>CoLab</i> . Observation atomique, c'est-à-dire localisée dans le temps. Elle est rattachée à un critère de grille dont le type est un ensemble d'observations. Les types d'observation peuvent être définis dans des catégories d'observation.
catégorie d'observation	<i>CoLab</i> . Les types d'observation de grille peuvent être regroupées en catégories dans une structure d'arbre distincte du patron de grille. Cette structure permet de regrouper des observations atomiques dans une typologie. Par exemple, on pourrait avoir la catégorie « erreur » qui comprend les observations de type « omission », « égarement », « substitution », « addition » et « abandon ». À noter qu'une observation peut avoir plusieurs types.

- texte libre (une longueur de caractères maximale peut être définie);
- nominal libre (un sous-ensemble de valeurs textuelles parmi un ensemble fini, un minimum et un maximum peuvent être définis);
- nominal restreint (exactement une valeur textuelle parmi un ensemble fini défini);
- quantitatif (un nombre, auquel peut être associée une unité du système international).
- temporel (une durée).

TABLEAU 3.8 – Comparatif

Méta-modèle	<i>décomposition</i>	<i>atomicité</i>	<i>projection</i>	<i>séquentialité</i>	<i>opérativité</i>
HTA	✓	✗	✓	✗	✗
TSK	✓	✓	✓	✓	✓
GTA	✓	✓	✓	✓	✗
GOMS	✓	✗	✓	✗	✓
Ontologie des AVQ	✓	✗	✓	✓	?

Enfin, on notera que les observations peuvent être catégorisées dans une hiérarchie en forme d'arbre (catégorie d'observation).

3.5.2 Recherche d'un noyau aux méta-modèles du comportement

Il n'y a, à notre connaissance, ni consensus, ni modèle comportemental formel recouvrant toutes les disciplines apparentées à l'AC, comme les sciences cognitives, l'ergothérapie, l'éthologie, la psychologie, l'ergonomie, etc. La sémantique du comportement dépend donc de l'expertise appliquée et nous ne pouvons pas imposer aux utilisateurs de l'API SCÉNARII et de COLAB un modèle particulier. Une grille de lecture commune demeure l'analyse par tâche orientée vers les buts. On présuppose en effet que le sujet observé agit suivant des buts qu'il s'est formulés à lui-même. Dans la littérature, de nombreux métamodèles pour tâches sont décrits. Certains d'entre eux sont orientés-machines (AMBOSS, ASI/CEA). Nous retiendrons ceux qui sont orientés vers les buts, à l'instar de « Goals, Operators, Methods, and Selection (GOMS) » [42], « Hierarchical Task Analysis (HTA) » [3], « Task Knowledge Structure (TSK) » [43] et « Groupware Task Analysis (GTA) » [71]. Nous ajouterons à cette liste l'ontologie des AVQ proposée par CHEN et NUGENT [10]. Dans cette dernière, il faudra substituer « tâche » par « activité » pour obtenir une équivalence.

Plutôt que d'essayer de construire un « méta-métamodèle pour les unir tous » — tâche ardue s'il en est, nous cherchons un noyau de propriétés communes qui serviront d'une base suffisante pour l'analyse. À cette base on peut ajouter des propriétés optionnelles pour couvrir un large

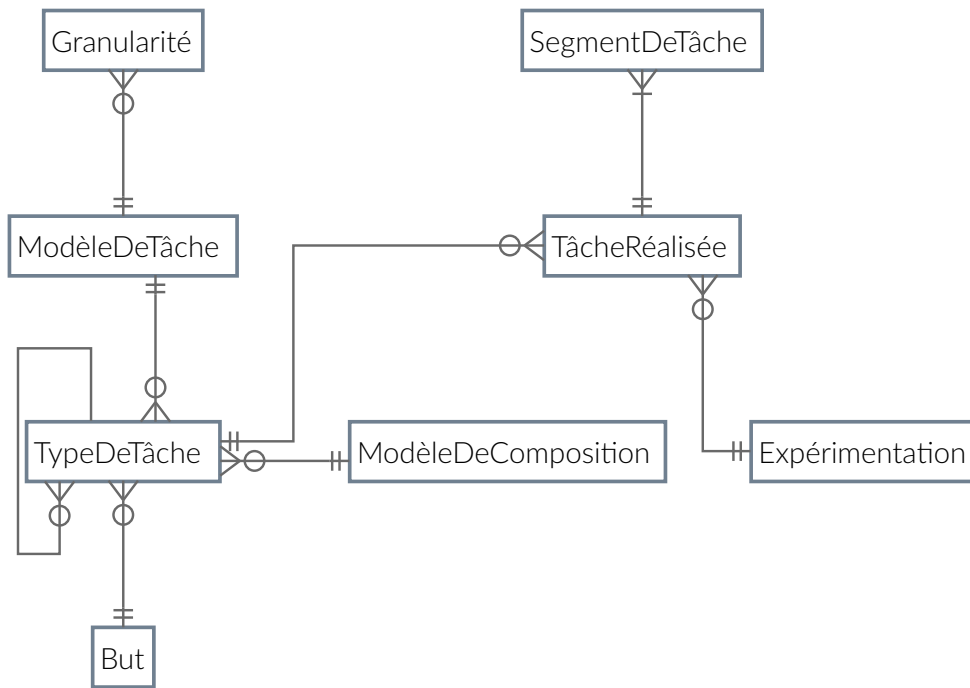


FIGURE 3.25 – Modèle de données logique des tâches

éventail de modèles existants. Dans ces cinq métamodèles que nous avons identifiés, on peut remarquer les éléments structurants suivant :

- *décomposition* : les tâches peuvent être constituées d'autres tâches ;
- *atomicité* : les actions représentent des tâches insécables ;
- *projection* : les tâches sont orientées vers des buts ;
- *séquentialité* : les tâches ont des pré et post-conditions qui les relient à d'autres tâches ;
- *opérativité* : on distingue les tâches, prescrites, de leur réalisation.

Le **Tableau 3.8** fait la synthèse des propriétés structurantes suivant ces cinq modèles. On peut observer que le noyau est constitué, à minima :

- de la décomposition ;
- de la projection.

À cette ébauche de noyau on doit ajouter les propriétés minimales relatives à une tâche, c'est-à-dire un ensemble de segments ayant chacun un « temps de début », « temps de fin » et, bien sûr, le sujet. Dans la **Figure 3.25**, le modèle de tâches est constitué d'un ensemble de types de tâches. Chaque type de tâche *peut* avoir un « parent » et plusieurs « enfants ». Dans ce dernier cas, un modèle de composition décrit l'ordonnancement et le parallélisme des sous-tâches. Si

n granularités sont définies pour le modèle, alors la hiérarchie formée par les types de tâches ne peut dépasser n niveaux d'arborescence. Une granularité définit aussi le nom des niveaux. On pourrait par exemple avoir un modèle comprenant trois granularités, « activité », « tâche » et « sous-tâche ». Enfin, l'atomicité est supportée de la manière suivante : la granularité la plus fine peut être définie comme atomique, auquel cas les segments de tâches réalisées dont le type de tâche est de granularité atomique n'ont pas de « temps de fin ».

3.5.3 Relation entre OACs et annotations

La Figure 3.26 décrit les relations entre les OACs, visibles au centre du diagramme, et les annotations, située à la périphérie (périmètre gris). Le chemin allant d'un outil à une annotation représente les critères de différenciation. On pourra remarquer que le critère inféré ne figure pas sur la ligne périphérique. La raison est qu'il n'a pas de saisie associée dans l'interface de l'atelier pour expérimentation, puisqu'il est issu d'un calcul sur d'autres annotations. On remarquera aussi qu'il existe des fonctions de conversion entre certaines annotations. Notamment, il est possible de convertir une observation de verbatim en observation de grille ou encore en début de segment de tâche réalisée. De plus, un critère inféré peut être lié à des propriétés issues d'autres critères, comme la durée de tâches réalisées, le nombre d'observation de grille d'un certain type ou encore des fonctions de critères quantitatifs.

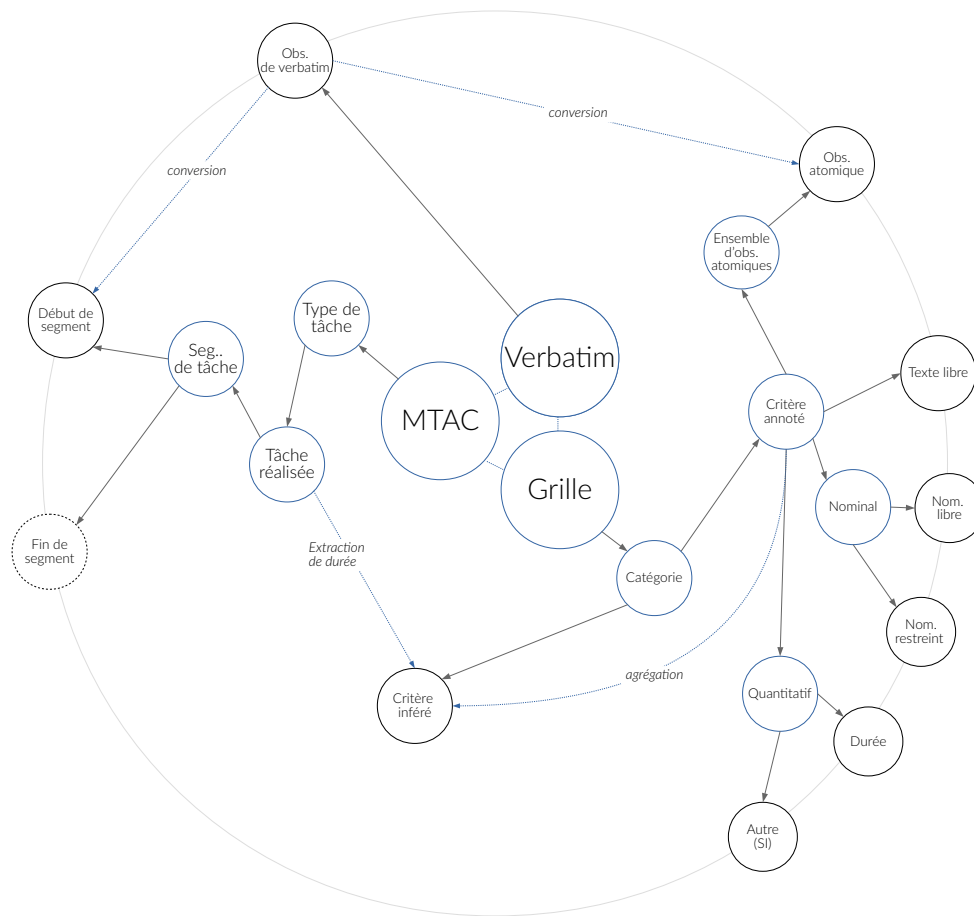


FIGURE 3.26 – Roue typologique des relations entre OACs et annotations

3.6 Modèle scénari

Dans la partie supérieure de la Figure 3.27, on retrouve la structure décrite en Figure 3.23. Dans la partie inférieure, on retrouve le modèle scénari. La correspondance est faite entre l'entité « Expérimentation » et « ScénarioAcquisition ». Ce dernier correspond à un **scénario d'enregistrement de données**. Il peut contenir des événements de capteurs, des capsules vidéos et enfin des **tâches réalisées**.

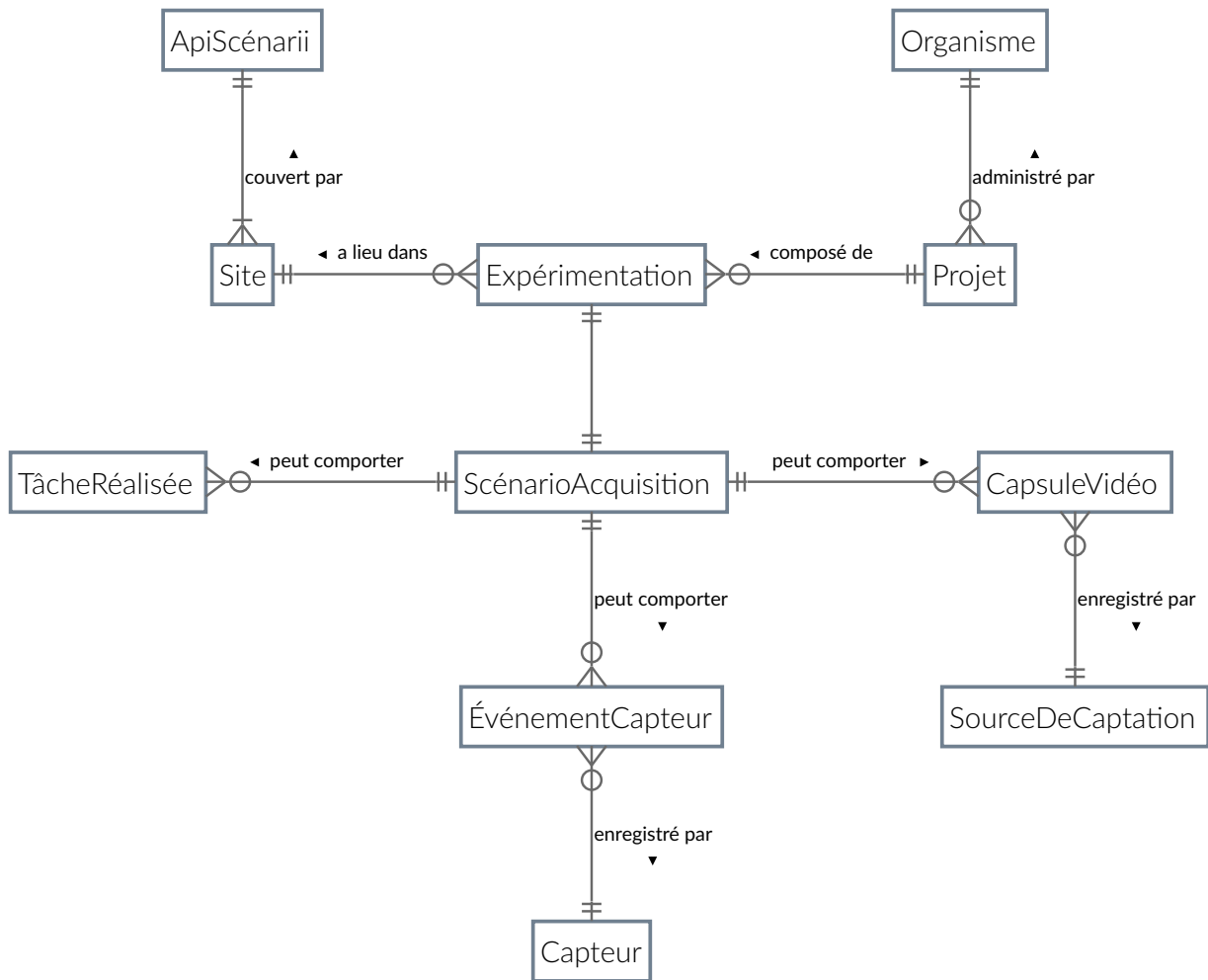


FIGURE 3.27 – Modèle logique de données scénarii

3.7 Identification des exigences de COLAB

Le **Tableau 3.9** fait l'inventaire des exigences¹¹ identifiées pour l'application COLAB. Ces exigences sont regroupées suivant les différents sous-systèmes identifiés dans la **section 3.3** et énumérés dans le **Tableau 3.2**.

11. Nous n'avons pas utilisé le mode conditionnel que recommande JACKSON car nous ne sommes pas dans un cadre contractuel avec un client.

3.7.1 Exigences fonctionnelles

TABLEAU 3.9 – Liste des exigences fonctionnelles de l’application CoLab

Réf.	Type	Description
LO1 – Système de contrôle d’accès		
<u>EXG_{CO} 001</u>	<i>Fonctionnalité</i>	Un utilisateur ne peut s’authentifier sur la plateforme qu’à l’aide d’un nom d’utilisateur ou d’une adresse courriel ainsi qu’un mot de passe.
<u>EXG_{CO} 002</u>	<i>Fonctionnalité</i>	Un futur utilisateur peut créer un compte sur la plateforme, en choisissant s’il le souhaite l’organisme pour lequel il s’affilie.
<u>EXG_{CO} 003</u>	<i>Fonctionnalité</i>	Un utilisateur doit être authentifié pour accéder aux actions que lui autorisent ses rôles
<u>EXG_{CO} 004</u>	<i>Contrainte de sécurité</i>	Un utilisateur doit avoir la responsabilité de collaborateur pour accéder à un projet et aux données d’expérimentation associées.
<u>EXG_{CO} 005</u>	<i>Fonctionnalité</i>	Un utilisateur peut demander à rejoindre un projet parmi les projets ouverts à l’inscription pour lesquels il n’est pas déjà inscrit.
LO5A – Atelier de configuration pour organisme		
<u>EXG_{CO} 006</u>	<i>Fonctionnalité</i>	L’ administrateur d’organisme doit pouvoir limiter l’inscription de nouveaux utilisateurs affiliés à son organisme à une liste blanche configurable comportant des adresses courriel et/ou des noms de domaines autorisés.
<u>EXG_{CO} 007</u>	<i>Fonctionnalité</i>	L’ administrateur d’organisme doit pouvoir inviter un collaborateur extérieur à son organisme.
<u>EXG_{CO} 008</u>	<i>Fonctionnalité</i>	L’ administrateur d’organisme peut associer des documents à un site d’ expérimentation , au format PDF, SVG, BMP ou JPEG.
LO5B – Atelier de configuration pour projet		
<u>EXG_{CO} 009</u>	<i>Fonctionnalité</i>	Un responsable de projet peut choisir d’ouvrir ou fermer les inscriptions à un projet .
<u>EXG_{CO} 010</u>	<i>Fonctionnalité</i>	Un responsable de projet peut paramétrer le projet pour accepter automatiquement des utilisateurs affiliés à l’ organisme en charge du projet.
LO5C – Atelier de configuration pour étude		
— suite à la page suivante —		

Liste des exigences fonctionnelles de l'application CoLab – suite

Réf.	Type	Description
<u>EXG_{Co} 011</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude doit pouvoir choisir le modèle de propriété des annotations , collectivisé par étude ou individualisé par associé d'étude .
<u>EXG_{Co} 012</u>	<i>Fonctionnalité</i>	Lorsqu'une étude est jugée terminée, c'est-à-dire que les annotations ne sont plus l'objet d'ajout ou de modification, le gestionnaire d'étude peut publier les résultats sur une page publique.
<u>EXG_{Co} 013</u>	<i>Fonctionnalité</i>	Un bouton sur la page de résultats d'étude publiés permet à un chercheur souhaitant réviser l'étude de demander à devenir visiteur d'étude .
<u>EXG_{Co} 014</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut choisir, pour un OAC déployé, d'activer les évaluations réflexives .
<u>EXG_{Co} 015</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude doit pouvoir mettre en place une feuille de route comportant des jalons.
<u>EXG_{Co} 016</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude doit pouvoir mettre en place des OACs issus du magasin libre des universités pour l'étude qu'il supervise.
<u>EXG_{Co} 017</u>	<i>Contrainte fonctionnelle</i>	Lorsqu'un gestionnaire d'étude choisit un OAC pour l'étude qu'il supervise, il a le choix entre garder l'outil tel quel — non modifiable — ou bien <i>forker</i> l'outil et transférer la propriété du nouvel outil au groupe d'étude.
<u>EXG_{Co} 018</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut choisir à tout moment de figer un OAC appartenant à l'étude, auquel cas l'outil ne peut plus être modifié par les associés d'étude .
<u>EXG_{Co} 019</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut choisir à tout moment de défiger un OAC appartenant à l'étude, auquel cas l'outil peut être modifié par les associés d'étude .
<u>EXG_{Co} 020</u>	<i>Contrainte fonctionnelle</i>	À la création d'une étude, le floutage d'expérimentation est activé.
<u>EXG_{Co} 021</u>	<i>Contrainte fonctionnelle</i>	Lorsque le floutage d'expérimentation est activé dans une étude, les associés d'étude n'ont accès qu'aux expérimentations floutées .
<u>EXG_{Co} 022</u>	<i>Contrainte d'utilisabilité</i>	À son premier accès à une étude, un collaborateur est introduit au concept d'expérimentation floutée.
<u>EXG_{Co} 023</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut à tout moment lever ou restituer le floutage d'expérimentation .

— suite à la page suivante —

Liste des exigences fonctionnelles de l'application CoLab – suite

Réf.	Type	Description
<u>EXG_{CO} 024</u>	<i>Fonctionnalité</i>	Un collaborateur peut créer une étude relative à un projet .
<u>EXG_{CO} 025</u>	<i>Contrainte fonctionnelle</i>	Le créateur d'une étude en est le gestionnaire par défaut.
<u>EXG_{CO} 026</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut transférer la gestion d'une étude à un associé.
<u>EXG_{CO} 027</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut muter un visiteur d'étude en associé d'étude.
<u>EXG_{CO} 028</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut accepter une demande d'accès pour visiteur . Si l' utilisateur qui souhaite rejoindre l' étude comme visiteur d'étude ne fait pas partie du projet , un responsable de projet doit aussi valider l'accès avant que le rôle ne soit attribué.
<u>EXG_{CO} 029</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut muter un associé d'étude en gestionnaire d'étude .
<u>EXG_{CO} 030</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut offrir une invitation à des collaborateurs pour qu'ils deviennent associés .
<u>EXG_{CO} 031</u>	<i>Fonctionnalité</i>	Un collaborateur peut accepter ou refuser une invitation à rejoindre une étude .
<u>EXG_{CO} 032</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut révoquer des associés d'une étude dont il a la charge, à condition qu'ils ne soient pas gestionnaires de cette même étude.
<u>EXG_{CO} 033</u>	<i>Contrainte fonctionnelle</i>	Pour devenir associé d'étude , un utilisateur doit être collaborateur du projet rattaché à l' étude .
<u>EXG_{CO} 034</u>	<i>Fonctionnalité</i>	Un gestionnaire d'étude peut mettre en place une évaluation avec accord interjuge.

LO2A – Assistant à la création d'OAC

<u>EXG_{CO} 035</u>	<i>Fonctionnalité</i>	Lorsqu'un utilisateur commence à créer un outil, ce dernier est en mode brouillon et n'est donc pas accessible dans le magasin libre des universités .
<u>EXG_{CO} 036</u>	<i>Fonctionnalité</i>	Un utilisateur peut créer un patron de grille .
<u>EXG_{CO} 037</u>	<i>Fonctionnalité</i>	Un utilisateur peut créer un MTAC .
<u>EXG_{CO} 038</u>	<i>Fonctionnalité</i>	Un utilisateur peut créer une hiérarchie de catégorie d'observation.

— suite à la page suivante —

Liste des exigences fonctionnelles de l'application CoLab – suite

Réf.	Type	Description
<u>EXG_{CO} 039</u>	<i>Contrainte fonctionnelle</i>	Lorsqu'un utilisateur crée un OAC , il en a la propriété.
<u>EXG_{CO} 040</u>	<i>Fonctionnalité</i>	Un utilisateur peut offrir et révoquer des permissions d'édition sur un OAC dont il a la propriété à un autre utilisateur ou bien à un groupe d'étude.
<u>EXG_{CO} 041</u>	<i>Contrainte fonctionnelle</i>	Lorsqu'un utilisateur veut utiliser un outil dans une étude, il doit le publier dans le magasin libre des universités avec une license Creative Commons.
<u>EXG_{CO} 042</u>	<i>Fonctionnalité</i>	Un utilisateur peut parcourir et utiliser tous les OACs publiés dans le magasin libre des universités .

LO2B – Atelier d'AC pour expérimentation

<u>EXG_{CO} 043</u>	<i>Fonctionnalité</i>	Le statut d'une expérimentation doit être visible par tous les collaborateurs du projet associé.
<u>EXG_{CO} 044</u>	<i>Fonctionnalité</i>	Lorsque la fonctionnalité est supportée par l'environnement d'acquisition, l' analyste comportemental peut consulter une carte en 2D positionnant l'ensemble des capteurs sélectionnés pour le projet.
<u>EXG_{CO} 045</u>	<i>Fonctionnalité</i>	L' analyste comportemental peut changer de source vidéo ; lors du basculement, le curseur de temps doit rester le même.
<u>EXG_{CO} 046</u>	<i>Fonctionnalité</i>	L' analyste comportemental peut changer rapidement de source vidéo à l'aide de raccourcis clavier.
<u>EXG_{CO} 047</u>	<i>Fonctionnalité</i>	L' analyste comportemental peut mettre en marche/arrêt la source sélectionnée à l'aide de la barre d'espace.
<u>EXG_{CO} 048</u>	<i>Fonctionnalité</i>	L' analyste comportemental peut changer la vitesse de lecture de la vidéo en $\times 0,5$; $\times 1$; $\times 2$ et $\times 4$.
<u>EXG_{CO} 049</u>	<i>Fonctionnalité</i>	Lorsque le curseur d'un analyste comportemental survole la barre d'avancement du lecteur vidéo, une image correspondant à une capture toutes les 5s autour de la valeur de temps survolée doit apparaître sous la forme d'une petite vignette.
<u>EXG_{CO} 050</u>	<i>Fonctionnalité</i>	Les tâches associées à une expérimentation doivent être visibles par l' analyste comportemental dans une fresque temporelle appelée « explorateur temporel ».

— suite à la page suivante —

Liste des exigences fonctionnelles de l'application CoLab – *suite*

Réf.	Type	Description
<u>EXG_{CO} 051</u>	<i>Fonctionnalité</i>	Un survol avec le curseur de souris dans l' explorateur temporel doit laisser apparaître un curseur de temps fantôme avec une indication du temps relatif.
<u>EXG_{CO} 052</u>	<i>Fonctionnalité</i>	Il doit être possible d'ajouter facilement un signet .
<u>EXG_{CO} 053</u>	<i>Fonctionnalité</i>	Un signet peut être converti en observation de grille .
<u>EXG_{CO} 054</u>	<i>Fonctionnalité</i>	Un signet peut être converti en observation de verbatim .
<u>EXG_{CO} 055</u>	<i>Fonctionnalité</i>	Un signet peut être converti en début de segment de tâche.
<u>EXG_{CO} 056</u>	<i>Fonctionnalité</i>	Un topic de l'espace de discussion pour étude peut être créé à partir d'un signet . Une référence au moment de l'expérimentation est alors insérer dans le corps du texte du message.
<u>EXG_{CO} 057</u>	<i>Fonctionnalité</i>	Lorsqu'un analyste comportemental survole une tâche de l'explorateur temporel avec le curseur, une infobulle doit apparaître donnant le détail de cette tâche.
<u>EXG_{CO} 058</u>	<i>Fonctionnalité</i>	Lorsqu'un analyste comportemental produit une annotation à partir d'un outil pour lequel les évaluations réflexives sont activées, il est invité à donner une note réflexive .
<u>EXG_{CO} 059</u>	<i>Fonctionnalité</i>	Un analyste comportemental doit pouvoir lancer la source vidéo courante au moment correspondant à un début de segment de tâche.
<u>EXG_{CO} 060</u>	<i>Fonctionnalité</i>	Un analyste comportemental doit pouvoir lancer la source vidéo courante deux secondes avant une observation de grille .
<u>EXG_{CO} 061</u>	<i>Fonctionnalité</i>	Un analyste comportemental doit pouvoir lancer la source vidéo courante deux secondes avant une observation de verbatim .
<u>EXG_{CO} 062</u>	<i>Fonctionnalité</i>	Lorsque la fonctionnalité est supportée par l'environnement d'acquisition, un analyste comportemental doit pouvoir consulter dans l'explorateur temporel, pour chaque source vidéo, un diagramme indiquant la présence ou pas de mouvement dans la vidéo.
<u>EXG_{CO} 063</u>	<i>Fonctionnalité</i>	Lorsque la fonctionnalité est supportée par l'environnement d'acquisition, un analyste comportemental doit pouvoir consulter dans l'explorateur temporel, pour un ensemble sélectionné de capteurs, les événements associés à ces mêmes capteurs.
<u>EXG_{CO} 064</u>	<i>Fonctionnalité</i>	Un analyste comportemental peut consulter les observations de verbatim dans une trame temporelle adjacente à la vidéo.

— suite à la page suivante —

Liste des exigences fonctionnelles de l'application CoLab – *suite*

Réf.	Type	Description
<u>EXG_{CO} 065</u>	Fonctionnalité	Un analyste comportemental peut éditer les observations de verbatim dans une trame temporelle adjacente à la vidéo.
<u>EXG_{CO} 066</u>	Fonctionnalité UX	Si le module « verbatim » est déployé dans l' étude en cours, dans l'atelier d' AC pour expérimentation , une combinaison de touche CTRL+ENTER réalise l'ensemble des actions suivantes : <ul style="list-style-type: none"> — mise en pause de la vidéo si elle était en lecture ; — affichage de la trame temporelle s'il était caché ; — lancement du formulaire de saisie d'observation de verbatim s'il n'était pas déjà ouvert ; — <i>focus</i> sur le formulaire de saisie d'observation de verbatim s'il n'était pas déjà <i>focusé</i>.
<u>EXG_{CO} 067</u>	Fonctionnalité	Les observations de verbatim doivent pouvoir être visibles en sous-titre des vidéos.
<u>EXG_{CO} 068</u>	Fonctionnalité	Un analyste comportemental peut créer une observation de grille à partir d'une observation de verbatim existante.
<u>EXG_{CO} 069</u>	Fonctionnalité	Un analyste comportemental peut créer un début de segment de tâche à partir d'une observation de verbatim existante.
<u>EXG_{CO} 070</u>	Fonctionnalité	Si un modèle de tâches est déployé dans une étude , un associé d'étude peut associer un critère de grille au temps d'exécution d'une tâche.
<u>EXG_{CO} 071</u>	Fonctionnalité	Un ensemble d' associés d'étude peut collaborativement alimenter les critères et catégories d'une grille d'évaluation .
Fonctionnalités transversales et sociales		
<u>EXG_{CO} 072</u>	Fonctionnalité	Chaque utilisateur doit pouvoir avoir rapidement accès à ses notifications et aux actions qui y sont liées.
<u>EXG_{CO} 073</u>	Fonctionnalité	Un utilisateur peut transformer certaines notification en tâches à réaliser.
<u>EXG_{CO} 074</u>	Fonctionnalité	Un utilisateur doit pouvoir accéder à un tableau de bord, dans lequel il a accès à ses études en cours, ses projets , l'atelier, le magasin libre des universités ainsi qu'à ses tâches à effectuer.
— suite à la page suivante —		

Liste des exigences fonctionnelles de l'application CoLab – *suite*

Réf.	Type	Description
<u>EXG_{CO} 075</u>	<i>Fonctionnalité</i>	Un associé d'étude a accès à l'espace de discussion relatif à l' étude dans laquelle il travaille.
<u>EXG_{CO} 076</u>	<i>Fonctionnalité</i>	Un collaborateur a accès à l'espace de discussion relatif au projet dans lequel il collabore.
<u>EXG_{CO} 077</u>	<i>Fonctionnalité</i>	Les opérateurs en domotique et administrateurs d'organisme ont accès aux espaces de discussion relatifs aux projets qui sont rattachés aux sites sous leur responsabilité.
<u>EXG_{CO} 078</u>	<i>Fonctionnalité</i>	Dans un message écrit sur un espace de discussion pour projet ou pour étude , un utilisateur peut se référer à une expérimentation avec la syntaxe suivante : #ANONYMOUS18 par exemple pour l' expérimentation floutée n°18.
<u>EXG_{CO} 079</u>	<i>Fonctionnalité</i>	Dans un message écrit sur un espace de discussion pour projet ou pour étude , un utilisateur peut se référer à un moment précis dans une expérimentation avec la syntaxe suivante : #ANONYMOUS18@13m18s par exemple pour le moment situé à 13 minutes et 18 secondes de l' expérimentation floutée n°18. <i>Confer EXG_{CO} 056 .</i>
<u>EXG_{CO} 080</u>	<i>Contrainte fonctionnelle</i>	Lorsqu'une expérimentation ou un moment d'expérimentation est référencé dans un message sur espace de discussion, un hyperlien vers cette expérimentation est généré dans le navigateur. Cet hyperlien pointe vers la route pour expérimentation d'étude dans le cas de l'espace de discussion pour étude, et vers la route pour expérimentation de projet dans le cas de l'espace de discussion pour projet.
<u>EXG_{CO} 081</u>	<i>Fonctionnalité</i>	Un utilisateur peut écrire un message personnel à un autre utilisateur à partir du profil de ce dernier.
<u>EXG_{CO} 082</u>	<i>Fonctionnalité</i>	Dans un message écrit sur un espace de discussion pour projet ou pour étude , un utilisateur peut se référer à un membre de son groupe (de projet ou d'étude) avec la syntaxe suivante : @nom-utilisateur.
<u>EXG_{CO} 083</u>	<i>Fonctionnalité</i>	Dans un message écrit sur un espace de discussion pour projet , un collaborateur ou opérateur en domotique peut se référer au groupe avec la syntaxe suivante : @projet.

— suite à la page suivante —

Liste des exigences fonctionnelles de l'application CoLab – *suite*

Réf.	Type	Description
<u>EXG_{Co} 084</u>	<i>Fonctionnalité</i>	Dans un message écrit sur un espace de discussion pour étude , un associé d'étude peut se référer au groupe avec la syntaxe suivante : @etude.
<u>EXG_{Co} 085</u>	<i>Fonctionnalité</i>	Lorsqu'un utilisateur est référé dans un message d'espace de discussion, directement ou par son groupe, il reçoit une notification.
LO2C – Éditeur de fonctions d'exploitation pour AC et LO3		
<u>EXG_{Co} 086</u>	<i>Fonctionnalité</i>	Un associé d'étude peut créer une fonction d'exploitation pour extraire les critères quantitatifs d'une grille d'évaluation .
LO4 – Module d'export		
<u>EXG_{Co} 087</u>	<i>Fonctionnalité</i>	Un associé d'étude peut exporter un verbatim ou un ensemble de verbatim au format PDF, DOCX et ODT.
<u>EXG_{Co} 088</u>	<i>Fonctionnalité</i>	Un associé d'étude peut exporter un formulaire de grille ou un ensemble de formulaire de grille au format PDF, XLSX, CSV et ODS.
ILCAG – ILC d'administration globale		
<u>EXG_{Co} 089</u>	<i>Fonctionnalité</i>	L' administrateur global peut mettre en place une politique de sécurité interdisant la réutilisation des X derniers mots de passe. <i>confer V2.25</i>
<u>EXG_{Co} 090</u>	<i>Fonctionnalité</i>	L' administrateur global peut configurer un nouvel organisme et son API SCÉNARII.

3.7.2 Exigences de conception et de performance

La **Tableau 3.10** recense les exigences de conception et de performance de l'application CoLAB.

TABLEAU 3.10 – Liste des exigences de conception et de performance de l'application CoLab

Réf.	Type	Description
Ciblage		
<u>EXG_{CO} 091</u>	<i>Contrainte architecturale</i>	L'application cliente doit être une application web de type Application Web Monopage (AWM) .
<u>EXG_{CO} 092</u>	<i>Contrainte de conception</i>	Au moment de la première livraison, l'application cliente doit être supportée par un ensemble navigateurs web représentant au minimum 90% des parts de marché au Canada sur environnement « desktop » (mobiles et tablettes sont donc ignorés), en excluant Internet Explorer.

3.7.3 Exigences de sécurité

La politique de sécurité (**Tableau 3.11**) est directement inspirée des recommandations de l'Open Web Application Security Project (OWASP) dans le standard **Application Security Verification Standard (ASVS)** (**Tableau 3.11**). Les contraintes de sécurité font face aux menaces les plus courantes répertoriées par l'OWASP, *Code Injection* (A1) *Cross-Site Scripting* (XSS, A2), *Broken Authentication and Session Management* (A3), *Broken Access Control* (A4), *Insufficient Attack Protection* (A7) et enfin *Underprotected APIs* (A10).

TABLEAU 3.11 – Liste des exigences de sécurité de l'application CoLab

Réf.	Description	ASVS
Sécurisation des navigateurs		
<u>EXG_{CO} 093</u>	Le serveur doit inciter le navigateur web à interdire les <iframe> en incluant systématiquement l'en-tête X-Frame-Options: SAMEORIGIN à ses réponses HTTP.	

— suite à la page suivante —

Liste des exigences de sécurité de l'application CoLab – suite

Réf.	Description	ASVS
<u>EXG_{Co} 094</u>	Le serveur web ne doit être disponible qu'en HTTPS, et il doit en informer le client en incluant systématiquement l'en-tête <code>Strict-Transport-Security: max-age=31536000; includeSubDomains; preload</code> à ses réponses HTTP.	V2.16

Sécurisation contre injection de code

<u>EXG_{Co} 095</u>	Les pages web livrées au client ne doivent pas inclure des ressources (images, scripts, polices, styles) hébergées en dehors du domaine du serveur.	
<u>EXG_{Co} 096</u>	Le serveur doit demander au navigateur web d'interdire l'exécution de scripts ainsi que le chargement de ressources qui ne proviennent pas du même domaine en incluant systématiquement l'en-tête <code>Content-Security-Policy: default-src 'self'</code> à ses réponses HTTP. <i>confer EXG_{Co} 095.</i>	
<u>EXG_{Co} 097</u>	L'API doit se prémunir contre des vulnérabilités de type « injection (No)SQL » en utilisant notamment du typage fort de paramètres, des schémas de validation restreints de ces mêmes paramètres et enfin des requêtes paramétrées en base de données.	V5.10, V5.20
<u>EXG_{Co} 098</u>	Les saisies utilisateur doivent avoir une longueur maximale et un schéma de validation.	V5.19, V5.21
<u>EXG_{Co} 099</u>	Protection XSS : Redéfinir	V5.19, V5.20
<u>EXG_{Co} 100</u>	Les saisies utilisateur doivent être validées, <i>sanitized</i> côté client et serveur.	V5.3, V5.5

Sécurisation de l'authentification

<u>EXG_{Co} 101</u>	L'accès à toutes les pages et à toutes les ressources — à l'exception des pages d'authentification et de création de compte — doivent requérir une authentification de l'utilisateur.	V2.1, V2.32
<u>EXG_{Co} 102</u>	Le mot de passe choisi par un utilisateur doit, pour être validé : — comporter au moins douze caractères ; — dont au moins un caractère spécial (non alpha-numérique ASCII) ; — dont au moins deux chiffres.	V2.7, V2.27

— suite à la page suivante —

Liste des exigences de sécurité de l'application CoLab – *suite*

Réf.	Description	ASVS
<u>EXG_{CO} 103</u>	Le formulaire de renouvellement du mot de passe doit contenir trois champs de saisie : l'ancien mot de passe, le nouveau mot de passe et la confirmation du nouveau mot de passe.	V2.9
<u>EXG_{CO} 104</u>	Le mot de passe doit être « salé » et « haché » avant d'être envoyé au serveur.	V2.9
<u>EXG_{CO} 105</u>	Lorsque l'authentification d'un utilisateur échoue, le message d'erreur ne doit pas exposer d'information sur l'existence ou non d'un compte associé à l'adresse courriel ou au nom d'utilisateur.	V2.18
<u>EXG_{CO} 106</u>	Un maximum de cinq tentatives d'authentification par minute doit être autorisé.	V2.20
<u>EXG_{CO} 107</u>	Les utilisateurs doivent pouvoir s'authentifier avec de multiples facteurs (2FA).	V2.31
<u>EXG_{CO} 108</u>	La gestion de session doit être faite avec une librairie éprouvée et doit se prémunir contre des attaques de type XSRF en utilisant, pour chaque transaction, des jetons anti-XSRF forts.	V4.13
<u>EXG_{CO} 109</u>	Le principe du moindre privilège (<i>principle of least privilege</i>) doit être garanti à partir d'un système de contrôle d'accès.	V4.1
<u>EXG_{CO} 110</u>	Le système de contrôle d'accès.	V4.1
<u>EXG_{CO} 111</u>	Les transactions à grand impact doivent nécessiter la saisie et la validation du mot de passe de l'utilisateur authentifié. (Liste des transactions :)	

Sécurisation de l'API (serveur)

<u>EXG_{CO} 112</u>	L'interface de programmation du serveur doit systématiquement s'assurer que les utilisateurs authentifiés ont les permissions nécessaires avant d'exécuter les opérations demandées ou accéder à des ressources confidentielles. <i>confer EXG_{CO} 108</i>	V2.4, V2.6, V4.4, V4.9
<u>EXG_{CO} 113</u>	Un « gouverneur de ressources » (<i>resource governor</i>) doit limiter le nombre d'opérations de changement d'état par un utilisateur par heure.	4.14

Journalisation

<u>EXG_{CO} 114</u>	Toute infraction à une police de sécurité doit être journalisée et reportée aux administrateurs-système.	
<u>EXG_{CO} 115</u>	Tout refus d'accès par le système de contrôle d'accès doit être journalisé.	V4.12

— suite à la page suivante —

Liste des exigences de sécurité de l'application CoLab – *suite*

Réf.	Description	ASVS
<u>EXG_{CO} 116</u>	Toute opération d'authentification et de modification des informations d'authentification doit être journalisée.	V2.12
<u>EXG_{CO} 117</u>	Les saisies d'utilisateur invalidées pour risque de XSS doivent être journalisées et doivent aboutir à un refus de requête.	V5.3

3.8 Caractéristiques fondamentales de API SCÉNARII

On se contentera ici de rapporter quelques caractéristiques essentielles, le projet n'étant pas suffisamment avancé pour aboutir à de véritables exigences.

3.8.1 Caractéristiques de l'API SCÉNARII

Pour autant, on souhaite que l'API SCÉNARII satisfasse les conditions suivantes :

- l'API doit suivre le style architectural **RE**presentational State Transfer (REST) ;
- l'API doit être sécurisée suivant les recommandations de l'**OWASP** ;
- l'API doit exposer le modèle scénarii ;
- un mécanisme de correspondance de modèles entre l'API et COLAB doit être conçu. Notamment, il faudra trouver un moyen pour que les scénarii d'acquisition soient synchronisés automatiquement avec un projet, à la suite d'une configuration mise en place par l'opérateur en domotique ou l'administrateur d'organisme ;
- l'API doit offrir un service vidéo similaire à medianode.

Aussi, une approche optative et modulaire est envisagée, pour permettre à chaque organisme de participer à la coopérative. Ainsi, une API SCÉNARII peut déclarer les modules et les sources de captation qu'elle supporte :

- sources vidéo ;
- capteurs d'environnement ;
- détection de mouvement par analyse vidéo ;
- reconnaissance d'activités avec **MTAC**.

Bien entendu, cette liste sera amenée à s'enrichir à mesure que le standard de l'API SCÉNARII évolue.

3.8.2 Contraintes environnementales

Aussi, l'environnement d'acquisition devra satisfaire les conditions suivantes :

- acquisition synchrone de toutes les sources ;
- la perte de synchronisation entre les sources d'acquisition ne doit pas excéder 100 ms ;
- il doit être possible pour l'opérateur en domotique de lancer un scénario d'acquisition.

Chapitre 4

Prototypage et validation

Dans ce chapitre, on présente les principales réalisations concrètes effectuées dans cette étude. D’abord, le prototype « COLAB 0.4 » (section 4.1). Ensuite, une étude ergonomique sur ce dernier (section 4.2). Puis, le prototypage de medianode (section 4.3). Enfin, la librairie `Serrurier` ainsi que le système de contrôle d’accès *Object Scope Delegate Role Based Access Control* (section 4.4).

4.1 Caractérisation du prototype COLAB 0.4

Cette section présente le prototype COLAB 0.4. Les captures d'écran sont disponibles en [Annexe E.1](#).

4.1.1 Choix de technologies et déploiement

COLAB 0.4 est une [AWM](#) développée à partir du cadriciel [Meteor](#). Ce cadriciel offre différents avantages qui en ont fait un excellent candidat pour le prototypage. C'est une solution *full stack* qui permet aux développeurs de se focaliser sur les niveaux « présentation » et « métier » de l'application riche. Cette solution permet notamment :

- de déléguer la communication entre les trois niveaux au cadriciel ;
- de bénéficier d'un partage de code-source entre la couche « présentation » et la « couche métier », on parle alors d'[isomorphisme](#) ;
- d'introduire de la [réactivité](#).

[Meteor](#) a beaucoup évolué depuis les débuts du projet [DEI](#) en 2014. Dans la version utilisée pour le prototype, il imposait MongoDB® — une technologie NoSQL — pour l'entreposage de données. La couche « métier » est déployée dans l'environnement d'exécution javascript node.js. Des informations plus détaillées sur l'architecture [Meteor](#) sont offertes dans la [section 4.4](#).

Dans les sections qui vont suivre, des captures d'écran seront fournies avec le chemin relatif depuis la racine — l'hôte de l'URL. Puisque la terminologie a évolué entre l'élaboration de ce prototype et la complétion de la phase d'analyse, les termes employés dans ce prototype ne reflètent pas toujours ceux définis dans le [dictionnaire univoque](#).

4.1.2 Considérations sur la sécurité

L'application web est accessible exclusivement par HTTPS, sur le réseau interne de l'Université. Les collaborateurs à l'extérieur doivent donc se connecter par VPN (Virtual Private Network) après avoir obtenu un compte invité conféré par l'Université. S'agissant de la gestion de mots de passe, la librairie `accounts-passwords` disponible pour [Meteor](#) et incorporée dans ce prototype utilise un algorithme de hachage SHA-256 pour le transport du client vers le serveur, ainsi que la fonction de hachage avec salage `bcrypt` pour l'entreposage. Ce modèle très robuste respecte la spécification V2.13 de l'[ASVS](#) relative au transport de mot de passe.

SECURITY REPORT

L'adresse courriel myecourriel@shouldnotwork.com n'apparaît pas dans la liste blanche.

WHEN : Mon Jun 19 2017 14:13:20 GMT-0400 (EDT)
USERID: unknown
IP : 10.44.161.13
USER-AGENT : Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:53.0) Gecko/20100101 Firefox/53.0
GEO :
 country_name: Canada,
 region_name: Quebec,
 city: Sherbrooke,
 zip_code: j1k 2r1,
 latitude: 45.3822,
 longitude: -71.9327

FIGURE 4.1 – Rapport d'activité suspecte expédié par courriel à l'administrateur-système

4.1.3 Fonctionnalités d'authentification

La page d'authentification (Figure E.1) est obligatoire pour accéder aux autres pages de l'application, exception faite de la création de compte (Figure E.2). Si un utilisateur souhaite accéder à une autre page alors qu'il n'est pas authentifié, il sera redirigé vers cette page. Lorsque les identifiants sont invalides, un message d'erreur apparaît : « L'identifiant ou le mot de passe est incorrect ». Au bout de dix tentatives par minute, l'authentification est bloquée pour l'adresse courriel ou le nom d'utilisateur, s'il existe, pour pendant une minute. Un rapport d'activité suspecte est alors généré et expédié aux administrateurs. Sur la page d'inscription (Figure E.2), un nouvel utilisateur peut créer un compte. Son mot de passe doit contenir « 8 caractères minimum et doit être constitué d'au moins : une lettre minuscule, une lettre majuscule ainsi qu'un chiffre ou un caractère spécial ». Si son mot de passe ne respecte pas cette contrainte, un message d'erreur est affiché (Figure E.3). De plus, une liste blanche a peut être activée par un administrateur. Si l'adresse courriel utilisée par l'individu qui souhaite s'inscrire n'y apparaît pas, un message « Accès prohibé » est affiché (Figure E.3). Un rapport d'activité suspecte est alors généré et expédié par courriel aux administrateur globaux (Figure 4.1).

Après l'étape d'authentification, l'utilisateur arrive sur la page du tableau de bord (Figure E.4). Depuis cette page, il peut rejoindre un projet (Figure E.5), ouvrir une page relative à un des projets dont il est membre (Figure E.6) ou encore changer de mot de passe.

4.1.4 Fonctionnalités relatives à un projet

Dans les captures qui vont suivre, le projet DEI sera utilisé comme exemple. Suivant la responsabilité de l'utilisateur, l'affichage de la page d'accueil d'un projet sera différente. Pour un



FIGURE 4.2 – Icônes de statut d'expérimentation

associé¹, une liste d'expérimentations anonymisées² apparaît (Figure E.6). Pour retirer l'anonymisation³, un utilisateur doit être muté responsable⁴. Les responsables de projet ont donc accès au détail des expérimentations, comprenant la classe du participant⁵ (Figure E.8). L'utilisateur peut utiliser le filtre textuel pour sélectionner un sous-ensemble d'expérimentations (Figure E.7, Figure E.9). Le responsable de projet a accès à un panneau de configuration du projet (Figure E.10), dont les fonctionnalités vont être détaillées dans le paragraphe ci-dessous. Les utilisateurs qui ne sont pas associés au projet sont redirigés vers leur page de tableau de bord s'ils essayent de joindre un chemin relatif au projet ainsi que tous ses sous-chemins, comme par exemple `projects/DEI` et `projects/DEI/rules`.

Les fonctionnalités détaillées dans ce paragraphe ne sont accessibles qu'aux responsables de projet. Un utilisateur non-autorisé qui voudrait accéder à une de ces pages se verra redirigé vers son tableau de bord `user/dashboard`. À noter que le design de ces pages est moins soigné car leur utilisation est très ponctuelle et qu'il s'agit d'un prototype.

Dans la page de gestion des membres (Figure E.11), le responsable de projet peut muter ou déclasser un membre. Dans la page de règles de gestion de projet (Figure E.12), le responsable peut configurer certaines options relativement aux autres utilisateurs. Dans la page de *template*, un responsable peut configurer les tâches (Figure E.13) et les annotations (Figure E.14) sous la forme d'une arborescence.

4.1.5 Fonctionnalités de visionnage et d'analyse d'expérimentation

Cette page concentre le cœur fonctionnel de l'application (Figure E.15). Pour comprendre l'interface plus en détail, on propose une capture légendée (Figure E.16). En haut se trouve la « barre d'informations », qui comprend le nom de l'expérimentation, anonymisé ou non suivant

1. « collaborateur » dans le dictionnaire univoque.

2. À peu de choses près, « expérimentation floutée » dans le dictionnaire univoque. La principale différence est qu'une expérimentation floutée ne devrait pas laisser apparaître la date.

3. « Lever le floutage » dans le dictionnaire univoque.

4. Ce n'est certainement pas acceptable dans une version exploitable de COLAB. Mais l'évaluation à l'aveugle ayant été supportée tardivement, cette fonctionnalité permettant de lever le floutage n'a pas pu être implémentée.

5. Le nommage des expérimentations s'est fait au départ sans tenir compte de la contrainte « aveugle », laissant apparaître la classe du participant. Un nom anonymisé a donc été ajouté *a posteriori*; ce n'est évidemment pas acceptable dans une version exploitable de COLAB.

la **responsabilité** de l'utilisateur, ainsi que le statut de l'expérimentation (Figure 4.2). Ce statut contient quatre icônes :

1. « publication » : est-ce que l'expérimentation est accessible à tous les **collaborateurs** ? Ce statut permet aux **opérateurs en domotique**⁶ de contrôler le moment de publication de l'expérimentation ;
2. « validation technique » : est-ce que l'expérimentation a été validée d'un point de vue technique ? Un **opérateur en domotique** documente d'éventuelles erreurs, et si aucune acquisition n'a pu être faite, il invalide l'expérimentation⁷ ;
3. « validation clinique » : est-ce que l'expérimentation a été validée d'un point de vue clinique ? Un **responsable de projet** peut invalider une expérimentation si un événement relatif au **participant** a empêché son bon déroulement⁸ ;
4. « synchronisation vidéo » : est-ce que les vidéos de l'expérimentation ont été synchronisées ? Un **opérateur en domotique** valide la synchronisation une fois que toutes les vidéos ont été synchronisées⁹.

Les différents contrôles effectués sur ces statuts sont accessibles dans l'onglet de collaboration (Figure 4.6, d). À gauche se trouve le « menu contextuel » (Figure 4.6). Par défaut, l'onglet de sélection de sources est affiché. Il permet de basculer entre les différents points de vue capturés durant l'expérimentation. Pour une source donnée, un survol sur une de ces icônes in-

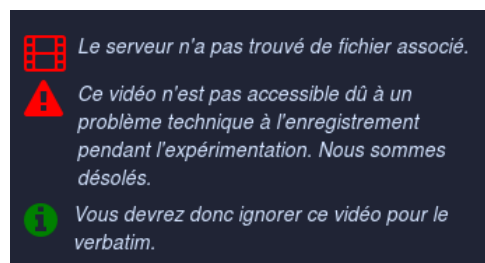


FIGURE 4.3 – Infobulle informative sur source de captation vidéo

6. Dans ce prototype, il n'y a pas de distinction entre un **responsable de projet** et un **opérateur en domotique**, on utilise donc ce terme ici dans le sens des objectifs-tâches associés plutôt que dans le sens de la responsabilité.

7. Nous avons eu une expérimentation invalidée de cette manière, pour cause de panne de courant dans l'Université.

8. Nous avons eu une expérimentation invalidée de cette manière, car le participant a subi un malaise.

9. Plusieurs explications à cette non-synchronicité. La première est que toutes les sources vidéos n'étaient pas raccordées au système d'acquisition vidéo. La deuxième est qu'un délai de quelques secondes était observable entre le début d'acquisition horodaté par le script `ipcamsh` et les premières images enregistrées dans la vidéo. Ce défaut de spécifications sur le dispositif d'acquisition a en partie motivé la spécification de API SCÉNARII.

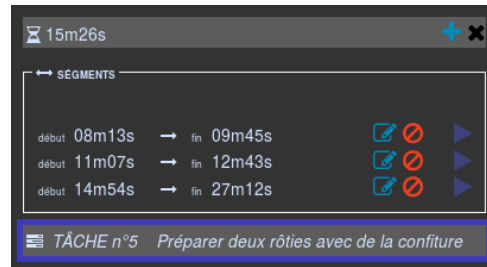


FIGURE 4.4 – Infobulle informative sur les segments de tâche

formatives laisse apparaître une infobulle (Figure 4.3). Ces informations ont été saisies par un opérateur en domotique ¹⁰.

Les touches 1 à 6 offrent un raccourci pour faire la bascule. Au milieu, l'« afficheur vidéo » présente la source actuellement sélectionnée. À droite figurent un interrupteur temps relatif / temps absolu ainsi qu'une interface de contrôle de l'explorateur temporel. En dessous, la « barre de contrôles » permet de créer une nouvelle annotation, de cacher l'explorateur temporel (Figure E.17), de lancer la lecture de la source sélectionnée ou de la mettre en pause, de changer la vitesse de lecture ($\times 1$, $\times 2$, $\times 4$), modifier le niveau sonore et enfin se déplacer sur l'axe temporel de la vidéo.

Tout en bas, l'« explorateur temporel » affiche les tâches codées par un utilisateur de l'application Android APD (Application Projet Dépistage) développée par un collègue du laboratoire, Mathieu Gagnon ¹¹. Leur couleur est définie par un responsable de projet (Figure E.13). À chacune de ces tâches est associé un ensemble de segments, qui sont des portions de temps pendant lesquelles la tâche est activement poursuivie par le participant. Un clic sur un segment de tâche permet de visualiser les informations relatives à cette tâche, et éventuellement les modifier (Figure 4.4). Enfin, un clic sur une annotation fait apparaître une infobulle permettant d'identifier

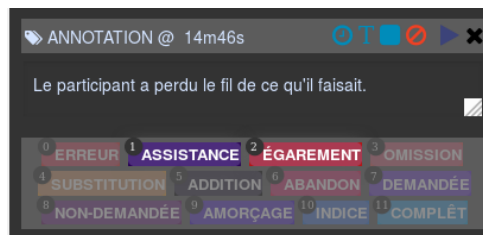


FIGURE 4.5 – Infobulle informative sur annotation

10. L'opérateur devait renseigner un fichier YAML situé dans le même répertoire que le fichier vidéo. Confer section 4.3 pour plus de détails.

11. Nous avons créé un script *ad-hoc* pour importer ces données. Dans une prochaine itération, le successeur de l'APD sera compatible avec l'API SCÉNARII, ne nécessitant pas ce genre de manipulation.



FIGURE 4.6 – Les quatre onglets du menu contextuel de la page d'expérimentation

les catégories associées, ainsi que le texte descriptif (Figure 4.5).

4.1.6 Fonctionnalités d'administration

Les fonctionnalités ici détaillées ne sont accessibles qu'aux **administrateur globaux**. Un utilisateur non-autorisé qui voudrait accéder à une de ces pages se verra redirigé vers son tableau de bord `user/dashboard`. À noter que le design de ces pages est moins soigné car leur utilisation est très ponctuelle et il s'agit d'un prototype.

Dans la page d'administration des règles (Figure E.18), l'**administrateur global** peut définir des règles à portée globale. Dans la page d'administration des utilisateurs, il peut définir une liste blanche d'adresses courriel autorisées à l'inscription (Figure E.19) ainsi que muter d'autres utilisateurs en administrateurs. Enfin, dans la page d'administration du serveur vidéo (Figure E.21), il peut configurer l'URL du serveur vidéo ainsi qu'activer ou désactiver le module vidéo. D'autres configurations sont possibles à partir d'un fichier de configuration distribué avec l'application. Il est notamment possible de configurer le serveur de courriels ainsi que certaines options de sécurité.

4.2 Mesures ergonomiques sur CoLAB 0.4

Usability and joy of use might be partially incompatible, because the former requires consistency and simplicity, whereas the latter requires surprise and a certain amount of complexity.

— M. HASSENZAHL, A. BEU et M. BURMESTER [34]

L'outil CoLAB vise l'efficacité et l'efficience avant le plaisir d'utilisation. En d'autres termes, la dimension ludique de l'interaction avec le logiciel ne doit pas interférer avec les objectifs de travail de ses utilisateurs. Pour autant, leur bien-être est une conséquence souhaitable, car sa caractérisation serait la marque d'un fort potentiel d'adhésion. Le parti-pris esthétique de l'interface peut être ainsi résumé : sobriété, élégance et intelligibilité ¹².

Un formulaire d'utilisation ([Annexe F.1](#)) est structuré en plusieurs parties, chacune construite pour une exploitation particulière ([Tableau 4.1](#)). Deux parties (I et III) sont tournées vers la validation de pertinence fonctionnelle et la prospection fonctionnelle tandis que la seconde est orientée vers les qualités subjectives perçues par l'utilisateur. Pour bien comprendre la distinction, on demande à l'utilisateur comment il évalue l'*utilité* de certaines fonctionnalités existantes ou prospectées dans les parties I et III, comment il *perçoit* le logiciel globalement dans la partie II. Cette dernière a été réalisée à partir du questionnaire standardisé *AttrakDiff 2*, introduit par [34]. On se référera à la [section 2.5](#) pour connaître la démarche plus en détail.

4.2.1 Élaboration du questionnaire

Le ([Tableau 4.1](#)) présente la structure du questionnaire d'utilisation. Les questions référées dans la colonne « Questions » peuvent-être consultées dans l'[Annexe F.1](#).

Identification d'un profil d'utilisateur

L'identification du profil d'utilisateur est succincte mais obligatoire. Elle est couverte par les questions P.Q03–SI.Q03 du questionnaire d'utilisabilité de CoLAB ([Annexe F.1](#)). Ce profil va permettre, entre autres :

- de croiser les réponses aux questions relatives à l'utilité de certaines fonctionnalités avec les expertises (P.Q04), les compétences et les objectifs-tâche (SI.Q01–Q03) de l'utilisateur ;

12. L'objectif du présent travail est de confirmer l'existence de besoins justifiant la création de CoLAB, bien loin devant la validation d'un design d'interfaces qui, de toute manière, n'a pas été réalisé par un expert du domaine. Pour autant, nous assumons que nous cherchons à convaincre l'utilisateur en douceur, sans exubérance ni fantaisie.

TABLEAU 4.1 – Structure et exploitation du questionnaire d'utilisation de CoLAB

Section	Questions	Description	Exploitation
Préambule	P.Q01–Q04	Identification de l'utilisateur.	Extraction des domaines d'expertise. Mesure de maîtrise des outils numériques.
Partie I	SI.Q01–	Caractérisation des tâches réalisées par l'utilisateur.	Extraction des objectifs–tâches.
Partie I	SI.Q06–Q08	Appréciation qualitative de l'occurrence de bugs et de leur impact sur la qualité de travail.	Contrôle sur SI.Q01–Q05.
Partie I	SI.Q09–Q24	Validation de fonctionnalités du prototype et de choix de design.	Validation d'exigences fonctionnelles. Qualité pragmatique perçue.
Partie II	SII.Q01–Q28	Appréciation émotionnelle de l'outil.	Qualités hédoniques et pragmatiques perçues.
Partie II	SII.Q29	Mesure de « complaisance ». En effet, certains utilisateurs connaissent l'auteur de CoLAB.	Contrôle sur SII.Q01–Q28.
Partie III	SIII.Q01–Q05	Positionnement de CoLAB vis-à-vis de logiciels voisins.	Identification de concurrents.
Partie III	SIII.Q06–Q25	Validation du lexique propre au langage omniprésent et validation de besoins via des maquettes.	Prospection fonctionnelle. Confirmation du domaine métier.
Partie III	SIII.Q26	Question ouverte.	Informations bonus.

— de recroiser les réponses liées à l'utilisabilité avec l'âge et le niveau de maîtrise des outils numériques par l'utilisateur (P.Q03).

Évaluation de pertinence fonctionnelle

La mesure de pertinence – ou d'adéquation – fonctionnelle, permet de confirmer l'existence d'un ensemble de besoins qui justifie la création de l'outil. Contrairement aux mesures de qua-

lité perçue, elle se fait pour chaque exigence fonctionnelle de l'outil. Elle est couverte par les questions SI.Q09–Q24 du questionnaire d'utilisabilité de CoLAB ([Annexe F.1](#)). Deux types de fonctionnalités, incarnées par des interactions homme-machine, sont référencées. D'abord les fonctionnalités explicites, qui sont visibles dans l'interface. La question est alors formulée avec le motif suivant :

Motif de question pour évaluation de pertinence d'une fonctionnalité explicite

Avez-vous utilisé *<description de la fonctionnalité explicite>* :

- ☐ Oui, et ça m'a été très utile
- ☐ Oui, et ça m'a été plutôt utile
- ☐ Oui, mais ça ne m'a pas été utile
- ☐ Non, je n'en voyais pas en quoi ça pouvait m'être utile
- ☐ Non, je n'avais pas compris la nature de l'interaction avec cet élément graphique ou bien je ne l'avais pas remarqué

Ensuite, les fonctionnalités implicites, qui doivent être découvertes par l'utilisateur. La question est ici formulée avec le motif suivant :

Motif de question pour évaluation de pertinence d'une fonctionnalité implicite

Avez-vous utilisé *<description de la fonctionnalité implicite>* :

- ☐ Oui, et ça m'a été très utile
- ☐ Oui, et ça m'a été plutôt utile
- ☐ Oui, mais ça ne m'a pas été utile
- ☐ Non, je n'ai pas eu l'occasion de découvrir cette interaction

La principale différence est dans la nature des choix de réponse négatif. Pour une fonctionnalité explicite, deux scénarios négatifs sont possibles. Soit l'utilisateur a identifié l'élément d'interface mais il n'a pas jugé que ce serait utile pour lui, soit il ne l'a pas remarqué ou n'a pas compris la nature de l'interaction possible. Pour une fonctionnalité implicite, un seul scénario négatif est possible. Soit il a découvert l'interaction, auquel cas il choisit une parmi les trois réponses positives possibles, soit il ne l'a pas découvert.

Prospection fonctionnelle

La prospection fonctionnelle consiste quant à elle en la recherche de nouvelles fonctionnalités qui correspondraient à des besoins chez l'utilisateur. Elle est l'objet des questions

SIII.Q06–Q25 du questionnaire d'utilisabilité de COLAB ([Annexe F.1](#)). La question posée est construite avec le motif suivant :

Motif de question pour prospection fonctionnelle

Pour moi, la possibilité de *<description de la fonctionnalité>* serait ... :

Pas du tout utile
○ ○ ○ ○ ○ ○ ○ ○
Très utile

Le format de réponse est donc une échelle de Likert allant de -3 à $+3$. Ces questions ne sont jamais obligatoires, car l'utilisateur peut ne pas comprendre le sens de la fonctionnalité. Il ne doit pas juger de l'utilité absolue de la fonctionnalité, mais de l'utilité supposée vis-à-vis de ses besoins, ses objectifs-tâches. D'où l'insistance de partir de son cas particulier en introduisant la phrase avec une mise en contexte : « *Pour moi* ».

Mesure de qualité perçue avec *AttrakDiff 2*

Le questionnaire *AttrakDiff 2* couvre les questions SII.Q01–Q28 du questionnaire d'utilisabilité de COLAB ([Annexe F.1](#)). La version française, retenue dans le questionnaire ([Annexe F.1](#)), a été élaboré en 2015 par LALLEMAND et al. . Les différenciateurs sémantiques et leur groupe sont consignés dans le [Tableau 4.2](#). Lors de l'exploitation des résultats, la colonne « Opposé ? » permettra de sémantiser le score rapporté de chaque item, variant de -3 à $+3$.

TABLEAU 4.2 – Différenciateurs sémantiques du questionnaire d'utilisation

Question	Différenciateur sémantique	Groupe	Opposé ?
SII.Q01	Technique – Humain	QP	oui
SII.Q05	Compliqué – Simple	QP	oui
SII.Q08	Pas pratique – Pratique	QP	oui
SII.Q10	Fastidieux – Efficace	QP	non
SII.Q12	Imprévisible – Prévisible	QP	oui
SII.Q20	Confus – Clair	QP	non
SII.Q27	Incontrôlable – Maîtrisable	QP	non

— suite à la page suivante —

Différenciateurs sémantiques du questionnaire d'utilisation – *suite*

Question	Différenciateur sémantique	Groupe	Opposé ?
SII.Q02	M'isole – Me sociabilise	QH-I	non
SII.Q06	Amateur – Professionnel	QH-I	oui
SII.Q11	De mauvais goût – De bon goût	QH-I	oui
SII.Q13	Bas de gamme – Haut de gamme	QH-I	non
SII.Q14	M'exclut – M'intègre	QH-I	non
SII.Q15	Me sépare des autres – Me rapproche des autres	QH-I	oui
SII.Q16	Non présentable – Présentable	QH-I	non
SII.Q04	Conventionnel – Original	QH-S	oui
SII.Q18	Sans imagination – Créatif	QH-S	non
SII.Q22	Prudent – Audacieux	QH-S	oui
SII.Q23	Conservateur – Novateur	QH-S	oui
SII.Q24	Ennuyeux – Captivant	QH-S	non
SII.Q25	Peu exigeant – <i>Challenging</i>	QH-S	non
SII.Q28	Commun – Nouveau	QH-S	oui
SII.Q03	Déplaisant – Plaisant	ATT	oui
SII.Q07	Laid – Beau	ATT	non
SII.Q09	Désagréable – Agréable	ATT	oui
SII.Q17	Rebutant – Attirant	ATT	non
SII.Q19	Mauvais – Bon	ATT	oui
SII.Q21	Repoussant – Attrayant	ATT	non
SII.Q26	Décourageant – Motivant	ATT	oui

4.2.2 Résultats du questionnaire

Les réponses brutes et anonymisées au questionnaire sont fournies en [Annexe F.2](#). Quatre utilisateurs sur six ont répondu au questionnaire, les deux restants n'estimant pas avoir suffisamment utilisé l'outil pour en faire une évaluation. Aussi, les résultats présents dans ce rapport sont à prendre avec précaution, car ils sont tirés d'un échantillon très réduit. Pour autant, les

informations extraites de ces retours ont été précieuses pour valider certains besoins et évaluer l'utilité générale de l'outil.

Notation Dans les pages qui suivent, les valeurs données sans unité sont des valeurs sur une échelle de Likert variant de -3 à $+3$. Les résultats relatifs à des échelles de Likert sont donnés avec la syntaxe suivante :

$$[\textit{Terme 1} - \textit{Terme 2}] M = 2,00 ; \sigma = 1,00$$

avec M la moyenne sur l'échantillon d'utilisateurs et σ l'écart-type, arrondis à deux décimales. Deux échelles de Likert particulièrement fréquentes sont abrégées ainsi :

- [*Pas du tout commode* – *Très commode*] en [*PdtC* – *TC*];
- [*Pas du tout utile* – *Très utile*] en [*PdtU* – *TU*].

Mise en garde Ces moyennes sont à considérer avec beaucoup de précautions. En effet, l'échantillon est très restreint et la marge d'erreur n'a pas pu être calculée. Nous proposons cette moyenne pour offrir un aperçu de la répartition des réponses, mais nous ne prétendons pas que ces résultats sont représentatifs d'un utilisateur « moyen » de COLAB, si tant est qu'il existe.

Identification des profils d'utilisateur

Les utilisateurs sont jeunes ($M = 28,75$ ans; $\sigma = 4,27$ ans; P.Q02) et très confortables avec les outils numériques ($M = 2,50$; $\sigma = 0,50$; P.Q03). Trois utilisateurs ont une expertise orientée vers l'ergothérapie ou la gériatrie, et deux ont une expertise orientée vers l'analyse statistique. Un utilisateur a une expertise en programmation logicielle et en réseautique. Deux utilisateurs ont utilisé l'outil pour construire et appliquer une grille, deux pour rédiger un *verbatim* des participants, et enfin deux utilisateurs pour valider et interpréter les données de capteurs (SI.Q01).

Pertinence fonctionnelle

Dans ces résultats (Tableau 4.3), certaines questions n'ont pas été exploitées car elles concernaient des détails d'interaction (SI.Q16, SI.Q24).

TABLEAU 4.3 – Exploitation de la pertinence fonctionnelle rapportée

Exigence	Questions	Résultats	Interprétation
EXG _{CO} 045	SI.Q11	$[PdtU - TU] M = 2,25 ; \sigma = 0,75$	forte utilité
EXG _{CO} 046	SI.Q09	Deux utilisateurs ont compris l'interaction et l'ont trouvé « très utile », deux autres ne l'ont pas comprise.	forte utilité, visibilité faible
N/C	SI.Q10	Deux utilisateurs l'ont trouvé « très utile » et deux autres « plutôt utile ».	non exploitable
EXG _{CO} 047	SI.Q14	Quatre utilisateurs l'ont trouvé « très utile ».	forte utilité
EXG _{CO} 048	SI.Q15	Trois utilisateurs l'ont trouvé « très utile », un utilisateur « plutôt utile ».	forte utilité
EXG _{CO} 050	SI.Q17–Q18	$[PdtU - TU] M = 2,50 ; \sigma = 0,50$ $[PdtC - TC] M = 2,25 ; \sigma = 0,75$	très utile, assez commode
EXG _{CO} 057	SI.Q19–Q20	$[PdtU - TU] M = 1,50 ; \sigma = 1,25$ $[PdtC - TC] M = 2,5 ; \sigma = 0,50$	utilité moyenne, utilisabilité forte
EXG _{CO} 059	SI.Q21	Un seul utilisateur a découvert la fonctionnalité, et l'a jugé « plutôt utile ».	visibilité médiocre
EXG _{CO} 043	SI.Q23	Un utilisateur ne l'a pas trouvé utile, un autre l'a trouvé « très utile », un autre ne l'a pas remarqué et un dernier l'a trouvé « plutôt utile »	utilité très variable, trop grande visibilité
N/C	SI.Q12	$[Très lent - Très rapide] M = 2,00 ; \sigma = 1,00$ Un utilisateur (U1) a rencontré des problèmes de vitesse de chargement des vidéos.	Le VPN est probablement responsable.
N/C	SI.Q13	Les utilisateurs ont trouvé qu'il manquait un certain nombre d'informations.	

AttrakDiff 2

TABLEAU 4.4 – Réponses « sémantisées » au questionnaire d'utilisation

Question	Différenciateur sémantique	Dimension	Moyenne « sémantisée »	Écart-type
SII.Q01	Technique – Humain	QP	0	0,82

— suite à la page suivante —

Réponses « sémantisées » au questionnaire d'utilisation – *suite*

Question	Différenciateur sémantique	Dimension	Moyenne « sémantisée »	Écart-type
SII.Q05	Complicqué – Simple	QP	1,25	1,5
SII.Q08	Pas pratique – Pratique	QP	2,25	0,5
SII.Q10	Fastidieux – Efficace	QP	1,25	1,5
SII.Q12	Imprévisible – Prévisible	QP	2,5	0,58
SII.Q20	Confus – Clair	QP	1,5	0,58
SII.Q27	Incontrôlable – Maîtrisable	QP	2,25	0,5
SII.Q02	M'isole – Me sociabilise	QH-I	0,75	0,96
SII.Q06	Amateur – Professionnel	QH-I	2	0,5
SII.Q11	De mauvais goût – De bon goût	QH-I	2,25	0,5
SII.Q13	Bas de gamme – Haut de gamme	QH-I	1,75	0,5
SII.Q14	M'exclut – M'intègre	QH-I	0,75	0,5
SII.Q15	Me sépare des autres – Me rapproche des autres	QH-I	0,5	0,58
SII.Q16	Non présentable – Présentable	QH-I	2,5	0,58
SII.Q04	Conventionnel – Original	QH-S	1,5	0,58
SII.Q18	Sans imagination – Créatif	QH-S	1,75	0,96
SII.Q22	Prudent – Audacieux	QH-S	1,25	1,26
SII.Q23	Conservateur – Novateur	QH-S	1,5	1
SII.Q24	Ennuyeux – Captivant	QH-S	0,75	0,96
SII.Q25	Peu exigeant – <i>Challenging</i>	QH-S	0	0,82
SII.Q28	Commun – Nouveau	QH-S	1	0,82
SII.Q03	Déplaisant – Plaisant	ATT	2	0,82
SII.Q07	Laid – Beau	ATT	1,5	0,58
SII.Q09	Désagréable – Agréable	ATT	1,5	1
SII.Q17	Rebutant – Attirant	ATT	1,75	0,5
SII.Q19	Mauvais – Bon	ATT	2,5	0,58

— suite à la page suivante —

Réponses « sémantisées » au questionnaire d'utilisation – *suite*

Question	Différenciateur sémantique	Dimension	Moyenne « sémantisée »	Écart-type
SII.Q21	Repoussant – Attrayant	ATT	1,75	0,5
SII.Q26	Décourageant – Motivant	ATT	1,25	0,96

Les résultats relatifs à la section II ont été « sémantisés » dans le [Tableau 4.4](#). À chaque question est associée une échelle de Likert dont la valeur de réponse varie de -3 à $+3$. Pour obtenir la moyenne « sémantisée » de chaque question, on a donc fait la moyenne des réponses, puis on a pris la valeur opposée de cette moyenne pour les différenciateurs opposés ([Tableau 4.2](#)). Cela reste un premier indicateur suivant les quatre groupes définis dans *AttrakDiff 2*. Pour avoir une vue plus globale, une moyenne de chaque dimension *AttrakDiff 2* a été reportée dans la [Figure 4.7](#).

Discussion Les auteurs d'*AttrakDiff 2* considèrent que pour une dimension donnée, une moyenne entre 0 et 1 est « normale » et ne doit donc pas susciter d'inquiétude particulière. Une moyenne supérieure à 1 est un indicateur satisfaisant¹³. Les résultats sont donc encourageants, et notamment l'attractivité générale du produit (1,75). Ces indicateurs s'alignent avec les objectifs énoncés au départ, c'est-à-dire que l'efficacité et l'utilisabilité (QP) doivent prévaloir sur les composantes hédoniques (QH-I, QH-S). On retrouve ces résultats en analysant le détail des réponses, et notamment les valeurs aux extrémités du spectre des réponses. Si on s'intéresse ([Tableau 4.4](#)) aux valeurs strictement inférieures à 1, on peut isoler quelques tendances :

- les questions relatives au caractère social de l'application sont en retrait, mais pas préoccupantes pour autant (SII.Q02, SII.Q14, SII.Q15) ;
- le logiciel est jugé en partie « technique » et « challenging » (SII.Q01, SII.Q25).

S'il est regrettable que les aspects sociaux de l'application ne ressortent pas de ce questionnaire, ce n'est guère étonnant, puisque les principales fonctionnalités de collaboration n'étaient pas encore implémentées dans ce prototype. On peut donc s'attendre à une augmentation de la qualité hédonique-identité dans les versions à venir.

De l'autre côté du spectre, les valeurs supérieures ou égales à 2 soulignent les points forts de l'outil, qualifié par les utilisateurs de « bon » (SII.Q19), « plaisant » (SII.Q03), « pratique » (SII.Q08), « maîtrisable » (SII.Q27), « prévisible » (SII.Q12), « présentable » (SII.Q16), « professionnel » (SII.Q06) et « de bon goût » (SII.Q12). Les qualités pragmatiques de l'outil sont

13. Dans les termes des auteurs, « désirable ».

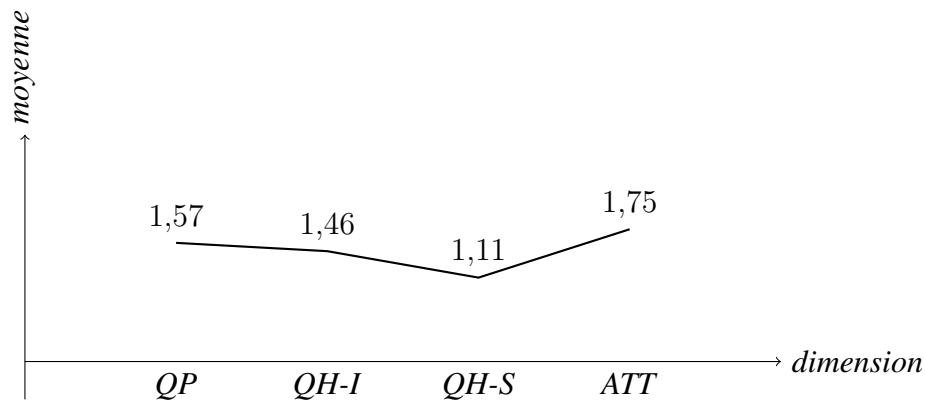


FIGURE 4.7 – Diagramme des valeurs moyennes des dimensions Attrakdiff

donc mises en avant par les utilisateurs, ainsi que son envergure professionnelle (QH-I) et son attractivité générale (ATT). La qualité hédonique-stimulation est en retrait, mais pas préoccupante pour autant. L'« originalité » et la « créativité » ne sont par exemple pas des qualités recherchées, et pourtant elles obtiennent un score largement satisfaisant (respectivement 1,5 et 1,75).

Fonctionnalités et interactions extraites de suggestions d'utilisateurs

Un certain nombre de fonctionnalités proposées par les utilisateurs se trouvent présentes dans la section sur la prospection fonctionnelle ; ces dernières ne sont donc pas mentionnées ici.

- il devrait être possible d'exporter des rapports à partir des annotations [U3, SII.Q03] ;
- lorsque le curseur de souris survole l'explorateur temporel, un curseur de temps fantôme devrait apparaître pour faciliter la navigation [U4, SI.Q22].

Validation de domaine

Ce questionnaire a permis de confirmer¹⁴ les définitions de *verbatim* (SIII.Q06) et *grille d'évaluation* (SIII.Q15).

Prospection fonctionnelle

L'exploitation des résultats est consignée dans le *Tableau 4.5*. Dans cette exploitation, les réponses vont être croisées avec le profil d'utilisateur, et notamment les tâches effectuées par ce dernier. En effet, ces tâches sont spécialisées, et les fonctionnalités imaginées ne sont pas

14. En excluant les réponses « sans avis »

nécessairement « utiles » pour tous les utilisateurs. Pour rappel, les questions relatives à la prospection fonctionnelle partent toujours du besoin de l'utilisateur interrogé, et pas des besoins que ce dernier pourrait projeter sur d'autres utilisateurs accomplissant des tâches distinctes des siennes.

Les questions sont toutes posées sur une échelle de Likert dont les termes sont *Pas du tout utile* (−3) et *Très utile* (+3). Les résultats sont donnés suivant deux échantillons. Le premier est l'échantillon cible, c'est-à-dire les utilisateurs dont les tâches-objectifs rejoignent la fonctionnalité proposée. Puis viennent les résultats sur l'échantillon total. Ils sont présentés avec la syntaxe suivante :

cible : $M = M_c$; $\sigma = \sigma_c$ [Ua, Ub, ...]

total : $M = M_t$; $\sigma = \sigma_t$

avec M_c la moyenne pour l'échantillon cible, σ_c l'écart-type pour l'échantillon cible, M_t la moyenne pour l'échantillon total, σ_t l'écart-type pour l'échantillon total et enfin « Ua, Ub, ... » désignent les utilisateurs de l'échantillon cible sélectionné. Dans le cas où la cible est équivalente à l'ensemble des utilisateurs, seul l'échantillon total est fourni.

TABLEAU 4.5 – Exploitation de la prospection fonctionnelle

Exigence	Question	Résultats	Interprétation
EXG _{CO} 065 – édition de verbatim à partir du fil temporel	SIII.Q08	cible : $M = 3,00$; $\sigma = 0,00$ [U3, U4] total : $M = 2,25$; $\sigma = 0,75$	très forte utilité
EXG _{CO} 067 – affichage des observations de verbatim en sous-titre des vidéos	SIII.Q09	cible : $M = 2,50$; $\sigma = 0,50$ [U3, U4] total : $M = 1,50$; $\sigma = 1,00$	forte utilité
EXG _{CO} 061 – lancement de la vidéo deux secondes avant une observation de verbatim	SIII.Q10	total : $M = 1,50$; $\sigma = 1,00$	utilité moyenne

— suite à la page suivante —

Exploitation de la prospection fonctionnelle – suite

Exigence	Question	Résultats	Interprétation
EXG _{CO} 062 – <i>affichage d'une détection de mouvement par source vidéo dans l'explorateur temporel</i>	SIII.Q11	total : $M = 2,5$; $\sigma = 0,75$	très forte utilité
abandonnée – <i>voir l'emplacement des sources vidéos sur une carte 2D</i>	SIII.Q12	total : $M = 1,75$; $\sigma = 0,88$	utilité moyenne
abandonnée – <i>voir plusieurs sources vidéo en même temps, sous la forme d'une mosaïque</i>	SIII.Q13	total : $M = 1,50$; $\sigma = 0,88$	utilité moyenne
EXG _{CO} 049 – <i>voir des vignettes lors d'un survol de la barre d'avancement du lecteur vidéo</i>	SIII.Q14	total : $M = 1,75$; $\sigma = 0,88$	utilité moyenne
EXG _{CO} 068 – <i>créer une observation de grille à partir d'une observation de verbatim</i>	SIII.Q18	cible : $M = 2,67$; $\sigma = 0,33$ [U1, U3, U4] total : $M = 2,00$; $\sigma = 1,00$	très forte utilité
EXG _{CO} 034 – <i>mener une évaluation inter-juge en groupe</i>	SIII.Q19	cible : $M = 2,67$; $\sigma = 0,33$ [U1, U3, U4] total : $M = 2,00$; $\sigma = 1,00$	très forte utilité
EXG _{CO} 070 – <i>utiliser la durée d'exécution d'une tâche comme critère de grille</i>	SIII.Q21	cible : $M = 1,67$; $\sigma = 0,88$ [U1, U3, U4] total : $M = 1,25$; $\sigma = 0,88$	utilité moyenne
EXG _{CO} 071 – <i>construire collaborativement une grille d'observation</i>	SIII.Q22	cible : $M = 2,00$; $\sigma = 1,00$ [U1, U3, U4] total : $M = 1,50$; $\sigma = 1,00$	utilité forte
EXG _{CO} 018 – <i>figer une grille d'évaluation</i>	SIII.Q23	cible : $M = 1,00$; $\sigma = 0,67$ [U1, U3, U4] total : $M = 0,75$; $\sigma = 0,75$	utilité modérée

— suite à la page suivante —

Exploitation de la prospection fonctionnelle – suite			
Exigence	Question	Résultats	Interprétation
EXG _{co} 063 – des événements de capteur sont visibles dans l'explorateur temporel	SIII.Q24	cible : $M = 1,00$; $\sigma = 0,67$ [U1, U3, U4] total : $M = 0,75$; $\sigma = 0,75$	utilité modérée
EXG _{co} 044 – voir une carte 2D de l'emplacement des capteurs sélectionnés	SIII.Q25	total : $M = 1,75$; $\sigma = 0,88$	utilité moyenne

Discussion Sans surprise, le score d'utilité perçue est systématiquement meilleur pour l'échantillon cible que pour le total des utilisateurs. C'est un indice que le ciblage de fonctionnalités est pertinent. À la suite de cette exploitation, deux fonctionnalités ont été abandonnées (SIII.Q12, SIII.Q13). Les autres ont été confirmées, et parmi les plus plébiscitées :

- EXG_{co} 065 : édition de verbatim à partir du fil temporel ;
- EXG_{co} 062 : affichage d'une détection de mouvement par source vidéo dans l'explorateur temporel ;
- EXG_{co} 068 : création d'une observation de grille à partir d'une observation de verbatim ;
- EXG_{co} 034 : évaluation inter-juge ;
- EXG_{co} 071 : construction collaborative d'une grille d'évaluation.

Ces résultats sont très satisfaisants, car les fonctionnalités prospectées que les utilisateurs jugent utiles sont au cœur de la solution COLAB.

4.2.3 Évaluation générale et synthèse

Les utilisateurs ont jugé l'outil beaucoup plus « nécessaire » qu'accessoire¹⁵ ($M = -2,25$; $\sigma = 0,44$; SI.Q02) en regard de leurs tâches. Les points positifs qui ressortent sont nombreux, et notamment l'expérience utilisateur est particulièrement satisfaisante avec des scores *AttrakDiff 2* au-dessus de la normale (Tableau 4.6). Ses qualités pragmatiques sont soulignées par les utilisateurs, qui le qualifient de « plaisant », « maîtrisable » et « prévisible ». De même, ils le trouvent « présentable », « professionnel » et « de bon goût » (QH-I). Un utilisateur (U1)

15. « Gadget » dans le questionnaire.

TABLEAU 4.6 – Synthèse des retours utilisateur

Positifs	Négatifs
— choix du point d’observation par l’analyste [SIII.Q02 : U1, U4]	— désynchronisation entre les vidéos [SIII.Q03 : U1, U3]
— annotations [SIII.Q02 : U3, U4]	— vidéos et son parfois absents [SIII.Q03 : U1]
— application disponible en ligne [SIII.Q02 : U2, U4]	— changement de source et chargement des vidéos lents [SIII.Q03 : U1]
— raccourcis clavier [SIII.Q02 : U2]	— modérément « technique » et « <i>challenging</i> » [AttrakDiff 2]
— esthétisme et simplicité / interface conviviale et sobre [SIII.Q02 : U2, U4]	— les fonctionnalités sociales se font attendre [AttrakDiff 2, SIII]
— plaisant, maîtrisable, prévisible, présentable [AttrakDiff 2]	

a considéré qu’il y avait eu un nombre non négligeable de bugs liés aux vidéos, mais il n’avait manifestement pas lu l’encadré adjoint qui explicitait que les erreurs liées au son et aux vidéos n’étaient pas des erreurs de l’application COLAB (SI.Q06, SI.Q07). Les problèmes liés aux vidéos concentrent cependant les critiques d’utilisabilité (Tableau 4.6). Mais COLAB en tant que tel n’était pas en cause¹⁶, et la prévalence de bugs a été évaluée à presque « aucun » par les trois autres utilisateurs $M = -2,33$; $\sigma = 0,44$; SI.Q06). L’impact de ces bugs sur le travail est relativement modéré [Nul – Très élevé] $M = -1,66$; $\sigma = 1,11$; SI.Q08.

4.3 Caractérisation du prototype medianode

Le serveur `medianode` est un serveur multimédia sécurisé. Implémenté avec NodeJS et Redis, il permet entre autre de contrôler l’accès aux vidéos d’*expérimentation* des *participants*. Mais sa conception est modulaire et configurable, si bien que cet outil pourra être utilisé dans d’autres projets. Ici, nous présenterons des éléments d’architecture et d’implémentation, mais

16. Deux obstacles ont été rencontrés. Le premier est l’utilisation du VPN, qui pouvait ajouter des délais à la lecture des vidéos. Le deuxième est le module d’acquisition vidéo (`ipcamsh`) qui n’offrait pas une synchronicité ni une fiabilité optimale à l’acquisition. Aussi, un environnement d’acquisition avec *scénarii d’acquisition* aurait permis d’aisément synchroniser les différentes données entre elles.

le détail du service et de son utilisation ainsi que la structure du fichier de configuration sont décrits dans la documentation fournie dans le dépôt Git¹⁷.

4.3.1 Structure et déploiement avec COLAB

Dans la [Figure 4.8](#), on présente comment `medianode` s’interface avec COLAB. Le fichier de configuration, en haut à gauche, contient d’abord la description du projet `DEI`, c’est-à-dire :

- l’emplacement de la racine dans l’arborescence de fichiers ;
- les règles de résolution de chemin pour atteindre les fichiers multimédias ;
- la compatibilité avec les fichiers de méta-information.

Ensuite, il contient la description de l’application COLAB, c’est-à-dire :

- les projets accessibles pour cette application ;
- l’activation des fichiers de méta-information ;
- la configuration du gestionnaire d’authentification.

Le gestionnaire d’authentification, visible à gauche du diagramme, est un module implémenté par l’utilisateur de la librairie qui vérifie les *credentials* du client qui souhaite s’authentifier. On se référera à l’[Annexe E.4](#) pour voir le module implémenté pour interfacer COLAB. On notera que dans cette implémentation, la fonction `isUserInProject` valide que l’utilisateur a les permissions nécessaires pour accéder aux vidéos d’expérimentation du `projet`. Le gestionnaire contacte donc l’entrepôt MongoDB pour confirmer la validité des *credentials* du client.

Depuis le client, les deux connexions — en gras — sont sécurisées (SSL). Dans un premier temps, il s’authentifie à COLAB puis, au serveur `medianode` avec les mêmes identifiants. Ce dernier renvoie en jeton dont la durée de validité dépend du fichier de configuration (dans notre cas, une heure). Avec ce jeton, le client peut accéder aux différents *endpoints* de l’API, parmi lesquels un service d’hébergement vidéo ([Tableau 4.7](#)).

17. Le dépôt est accessible à cette adresse : <https://github.com/sveinburne/medianode>.

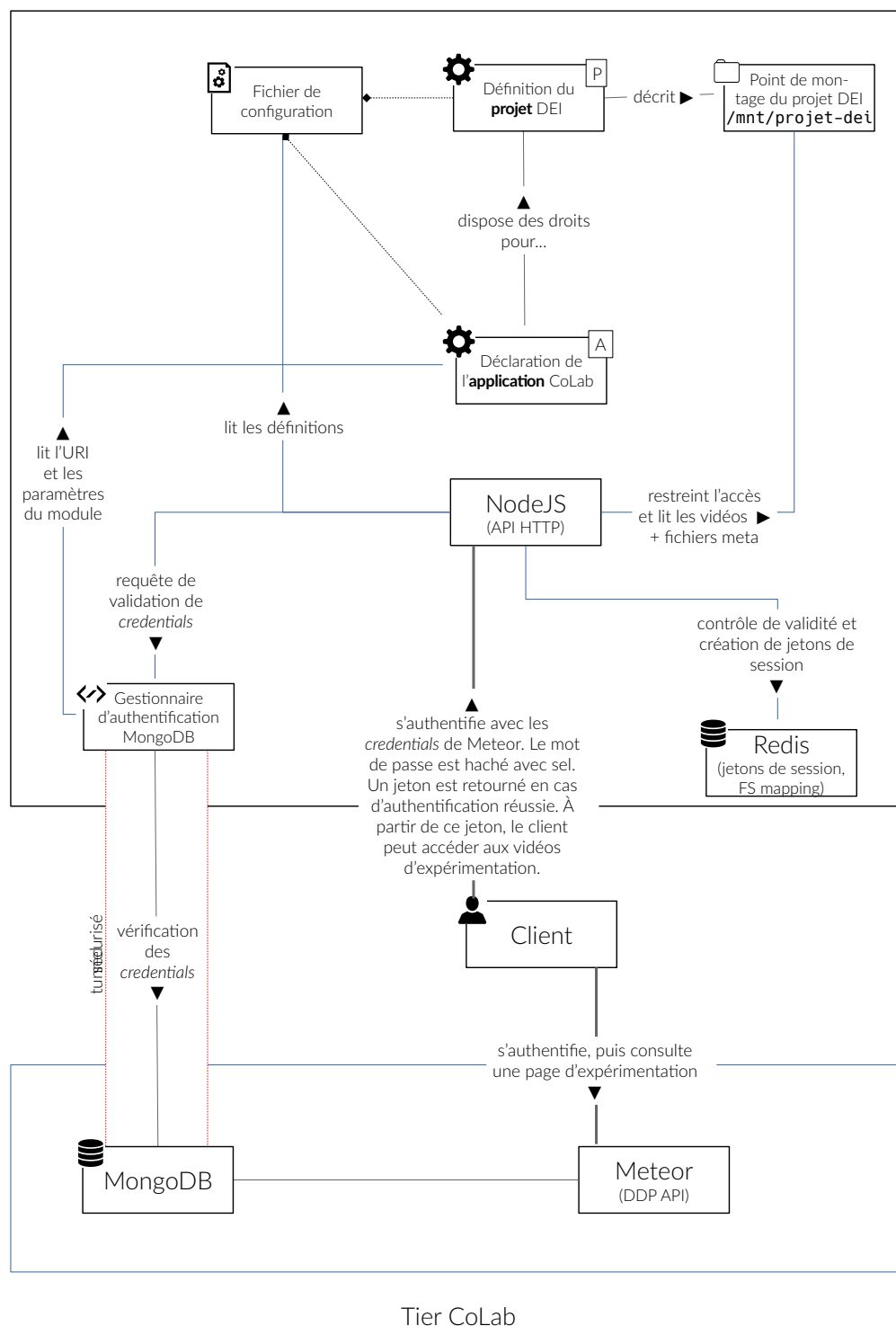


FIGURE 4.8 – Déploiement de medianode avec CoLab

4.3.2 Aperçu du service

La **Tableau 4.7** montre les différents *endpoints* de l'API. Si l'authentification est activée (`:disableAuth = false` dans le fichier de configuration), le client va devoir s'authentifier auprès du serveur vidéo.

```
1  {  
2    "places": [  
3      {  
4        "place": "chambre",  
5        "fileFound": true,  
6        "meta": {  
7          "warning": "Perte acq. à la 32'",  
8        }  
9      },  
10     {  
11       "place": "cuisine-plafond",  
12       "fileFound": true,  
13       "meta": {}  
14     }  
15   ]  
16 }
```

FIGURE 4.9 – Exemple de réponse à une requête `/i/:prj/:exp` (JSON)

Comme illustré dans la **Figure 4.8**, le mécanisme d'authentification est à la charge du développeur souhaitant utiliser le serveur vidéo. Il devra écrire un petit module en JavaScript de quelques lignes permettant de valider les identifiants du client.

Pour les deux derniers *endpoints*, l'identification se fait avec les paramètres de requête¹⁸ `a=:app` et `t=:token`. Aussi, un jeton est toujours associé à une application, et il est impossible d'utiliser un jeton pour une autre application que celle donnée en paramètre au moment de l'authentification.

Une réponse de la requête d'information est visible dans la **Figure 4.9**. Un champ `meta` donne des informations sur la qualité de la vidéo. Ces informations peuvent être jointes dans l'arborescence de fichiers du projet, dans un fichier YAML situé dans le même répertoire que la vidéo.

18. *Query string*, standard pour donner des paramètres à une URL.

TABLEAU 4.7 – *API service map*

Chemin	Méthode	description
<code>/auth/t/:app</code>	POST	Renouvellement d'un token sur l'application <code>:app</code> .
<code>/auth/c/:app</code>	POST	Authentification par identifiants sur l'application <code>:app</code> .
<code>/m/:prj/:exp/:src</code>	GET	Vidéo issue du point de captation <code>:src</code> durant l'expérimentation <code>:exp</code> associée au projet <code>:prj</code> .
<code>/i/:prj/</code>	GET	Ensemble des expérimentations disponibles pour le projet <code>:prj</code> .
<code>/i/:prj/:exp</code>	GET	Sources de capture pour l' <code>:exp</code> associée au projet <code>:prj</code> .

4.4 CoLAB 0.5 et Serrurier, proposition d'une librairie déclarative pour automatiser les contrôles d'accès par méthode

Le prototype CoLAB 0.5 a été l'occasion d'explorer certaines pistes de conception interne. Bien que l'objectif n'était pas spécialement le développement d'une interface graphique, des captures d'écran de ce prototype sont visibles en [Annexe E.2](#). Dans le cadre de ce prototypage, Serrurier [62] a été développée pour simplifier le contrôle d'accès¹⁹. Il est conçu pour fonctionner avec Astronomy [41], une librairie de gestion de modèles pour Meteor. Un exemple de classe définie avec Serrurier est accessible en [Annexe E.3](#).

4.4.1 Introduction à Astronomy et Meteor

La librairie Astronomy est définie comme suit par ses concepteurs : « The Astronomy package introduces the Model Layer for Meteor collections. It can also be named the Object Document Mapping system (ODM) or for people coming from relational database environments the Object-Relational Mapping system (ORM). »

Le système de persistance privilégié par Meteor est MongoDB, un système de base de données NoSQL orienté documents. La structure des données persistées n'est pas relationnelle mais

19. Le code source est accessible en *Open Source* (licence MIT) à l'adresse suivante : <https://github.com/sveinburne/serrurier>.

```

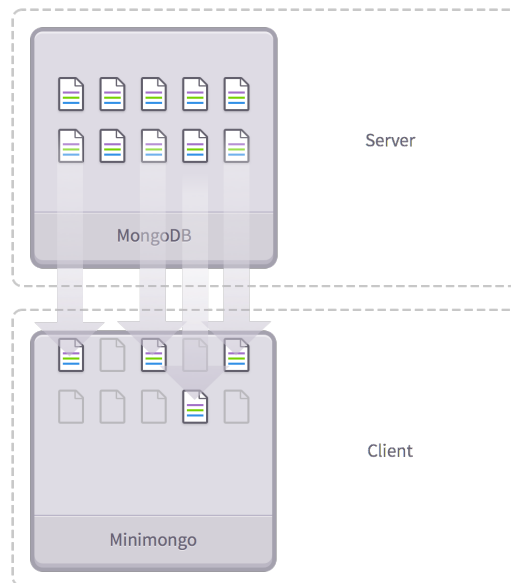
1  import { Class } from
   ↪ 'meteor/jagi:astronomy';
2
3  const Posts = new
   ↪ Mongo.Collection('posts');
4  const Post = Class.create({
5    name: 'Post',
6    collection: Posts,
7    fields: {
8      title: {
9        type: String,
10       validators: [{
11         type: 'minLength',
12         param: 3
13       }]
14     },
15     userId: String,
16     publishedAt: Date
17   },
18   behaviors: {
19     timestamp: {}
20   }
21 });

```

FIGURE 4.10 – Déclaration d’une classe dans Astronomy (ECMAScript 6)

imbriquée à la manière de JSON, et sans schéma contraignant. C’est pourquoi les auteurs font l’analogie avec un ORM, car Astronomy offre une approche déclarative des schémas associés aux collections.

Les collections sont les homologues des « tables » dans un système NoSQL, et Astronomy permet de leur appliquer des contraintes fines, comme des valeurs maximales et minimales pour

FIGURE 4.11 – Isomorphisme dans Meteor avec MongoDB – Minimongo et *pub – sub*

Source : GREIF, S., 2014, [29]

un champ donné. Les schémas sont déclarés de manière **isomorphe** comme des classes dans un contexte d'héritage par prototype. Un exemple minimal est visible en [Figure 4.10](#). L'argument de la méthode `create` est un objet de configuration déclarant les caractéristiques importantes de la « classe »²⁰. Parmi celles-ci, les `fields` définissent et contraignent les champs des instances de ce schéma. Dans l'exemple de la [Figure 4.10](#), le champ `title` est de type `String` et pour être valide, il doit être de longueur minimale « 3 ». Des méthodes peuvent aussi être définies à partir du champ de configuration `methods` (voir l'exemple complet en [Annexe E.3](#)). C'est sur celles-ci que `Serrurier` va s'appliquer. Avant de poursuivre, il est nécessaire de présenter l'architecture de `Meteor` 1.3. En effet, `Meteor` est un framework pour applications web **isomorphes** et **réactives**. Le mécanisme de synchronisation des collections, « Distributed Data Protocol » (DDP), est fondé sur le patron « publish – subscribe » ([Figure 4.11](#)).

Le serveur publie certaines collections, avec bien entendu des conditions de sécurité implémentées par l'utilisateur de la librairie, et le client s'abonne à des sélections — « curseurs » dans la terminologie de `Meteor` — qui sont des échantillons de ces collections ([Figure 4.13](#)). Dans cet exemple, le client s'abonne à deux sélections distinctes de la même collection. Le client détient donc une « vue » limitée de la base de données distante, et les changements d'état de ce modèle sont propagés par DDP, ce dernier étant construit sur le protocole Websocket. On remarquera que la propagation se fait « top-down », du serveur vers le client, et pas dans l'autre sens. Les mutations du modèle persisté souhaitées par le client peuvent se faire par deux mécanismes.

Soit par des *meteor methods*, un appel distant de méthode **isomorphe**. Dans ce cas, la modification est contrôlée par le serveur, ce qui permet de limiter considérablement la portée de la mutation et réduit les risques d'une brèche de sécurité. Soit par des requêtes MongoDB isomorphes, depuis le client vers le serveur. Cette dernière solution a l'avantage de permettre une mise à jour instantanée de la vue cliente sans délai apporté par le serveur (*Optimistic UI*), mais sa contrepartie est qu'elle nécessite de très bien penser la sécurité des opérations *Create*, *Update*, *Delete*. `Meteor` offre un certain contrôle sur ces opérations avec le patron « Allow-deny »

```

1 Messages.allow({
2   update: function (userId, doc, fields,
3     ↪ modifier) {
4     if (userId && doc.userId === userId) {
5       return true;
6     }
7   });

```

FIGURE 4.12 – Sécurisation d'une opération de mise à jour de collection dans `Meteor` (code serveur, ECMAScript 6)

20. Le mot-clé `class` existe aussi depuis ECMAScript 6. Dans *Astronomy*, il faut plutôt l'entendre comme synonyme de « schéma ».

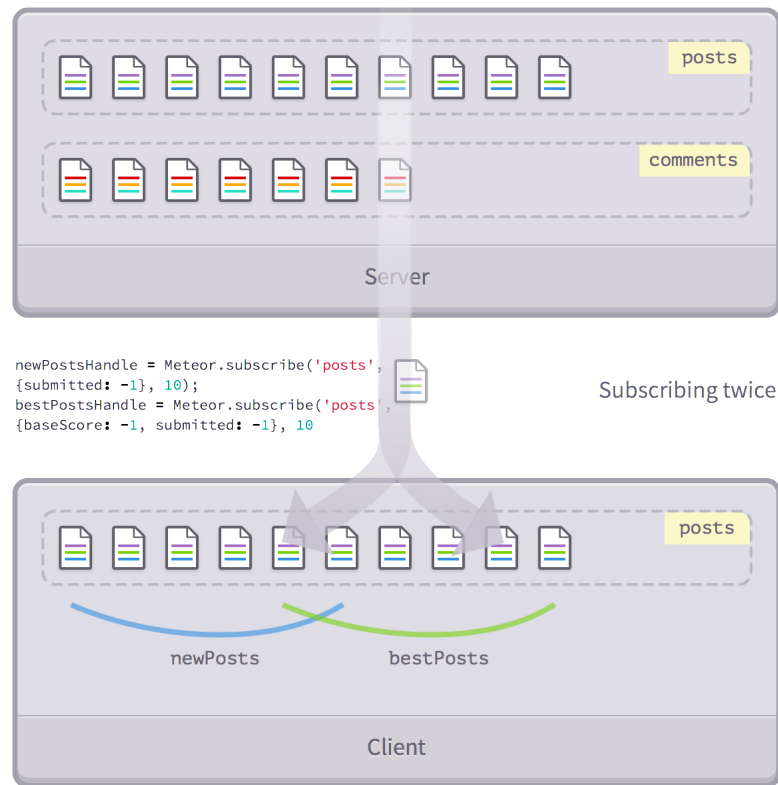


FIGURE 4.13 – Synchronisation réactive de deux sélections d’une même collection, avec DDP

Source : GREIF, S., 2014, [29]

(Figure 4.12). Astronomy apporte une solution satisfaisante aux problèmes de sécurité associés à cette dernière alternative en interdisant par défaut la mutation des objets, et en permettant d’une part de contrôler les mutations à partir de *callbacks* définis par l’attribut *events* de l’objet de configuration (Figure 4.10), d’autre part d’interdire la mutation de champs avec l’attribut de champ *immutable*.

4.4.2 Présentation des décorateurs en Javascript

Les décorateurs²¹ sont une proposition de stade 2 de la norme ECMAScript, mais leur support est déjà proposé par le transcompilateur Babel²² et TypeScript les offre nativement. Ils sont analogues aux attributs en C# et aux annotations en Java. Analogues mais bien distincts, car

21. Site statique de la proposition : <http://tc39.github.io/proposal-decorators/>.

22. En JavaScript, un transcompilateur compile notamment des versions d’ECMAScript encore partiellement supportées par les navigateur en des versions largement supportées. La librairie est accessible à l’adresse suivante : <https://babeljs.io/>.

```

1  function enumerable(value) {
2    return function (target, key, descriptor)
3      ↪ {
4        descriptor.enumerable = value;
5        return descriptor;
6      }
7  }
8  class C {
9    @enumerable(false)
10   method() { }
11 }

```

FIGURE 4.14 – Déclaration et application d’un décorateur (ECMAScript Next)

on les implémente avec des fonctions, et non des classes²³, de façon assez proche de Python. Un décorateur est une expression ayant la syntaxe suivante : `@mydecorator`(arg1, ..., argN). Ces expressions précèdent les déclarations sur lesquelles ils s’appliquent, ou bien une déclaration de classe, ou bien une déclaration de propriété de classe. Un exemple de déclaration et d’application sur propriété de classe est donné dans la Figure 4.14. Le décorateur est donc une fonction qui prend pour arguments les arguments passés à son expression — ici `false` — et qui retourne une fonction. Cette dernière fonction prend exactement trois arguments :

1. `target`, l’objet-prototype duquel seront dérivées les instances de classe ;
2. `key`, le nom de la propriété sur laquelle s’applique le décorateur ;
3. `descriptor`, le descripteur de propriété, introduit dans ECMAScript 5, qui contient des attributs de cette propriété (par ex., accesseurs ou valeur, lecture seule, lecture / écriture, énumérable et configurable).

Elle doit renvoyer ou bien le descripteur (même référence), ou bien la valeur de la propriété. Note importante : lorsque `Serrurier` a été implémenté, les bibliothèques existantes proposaient la version *stage-0* des spécifications. Entre temps, les spécifications sont passées en *stage-2*, mais les bibliothèques n’ont pas encore suivi.

4.4.3 Présentation de Serrurier

`Serrurier` va permettre, avec le décorateur `@cadenas`, de restreindre l’exécution des méthodes qu’il décore. On se référera à l’Annexe E.3.2 pour voir comment implémenter un `@cadenas`. Dans la Figure 4.15, le `@cadenas` `loggedUserInRole`²⁴ empêche l’utilisateur authentifié d’exécuter la méthode `updateSensitiveData` à moins qu’il soit administrateur. Il va de soi dans un contexte *isomorphe* que le contrôle effectué sur l’exécution de

23. En JavaScript, les fonctions peuvent être manipulées comme des objets.

24. Ce `cadenas` dépend de la bibliothèque de gestion de rôles `alanning:roles` accessible à cette adresse <https://github.com/alanning/meteor-roles/>.

```

1  import { Serrurier, cadenas } from 'meteor/svein:serrurier';
2  import { Mongo } from 'meteor/mongo';
3
4  const Project = Serrurier.createClass({
5    name: 'Project',
6    collection: new Mongo.Collection( 'projects' ),
7    methods: {
8      // This is it
9      @cadenas( 'loggedUserInRole', 'administrator' )
10     updateSensitiveData() {
11       // ...
12       console.info( 'The assertion passed, user is administrator.' );
13     }
14   }
15 });

```

FIGURE 4.15 – Utilisation du décorateur « cadenas » lors de la déclaration d’une « classe » Astronomy (ECMAScript Next).

cette méthode est doublé sur le serveur, le client étant jugé « peu fiable » (*double security check*). La Figure 4.16 illustre cette caractéristique pour la méthode `updateSensitiveData` proposée en Figure 4.15. Lorsque le cadenas détecte une violation, il lève une exception qui empêche la méthode d’être exécutée. Il notifie ensuite des rapporteurs sur écoute qui peuvent, par exemple, faire de l’audit de journal ²⁵.

4.4.4 Perspectives de conception interne : Object Scope Delegate Role Based Access Control

La version actuelle de Serrurier est limitée par la librairie de rôles utilisée. On propose ici de décrire une structure de rôles qui s’adapterait particulièrement bien à l’approche déclarative de Serrurier. Comme nous le faisons remarquer dans la section 1.4, le modèle proposé par FERRAILOLO, KUHN et CHANDRAMOULI laisse libre la résolution des per-

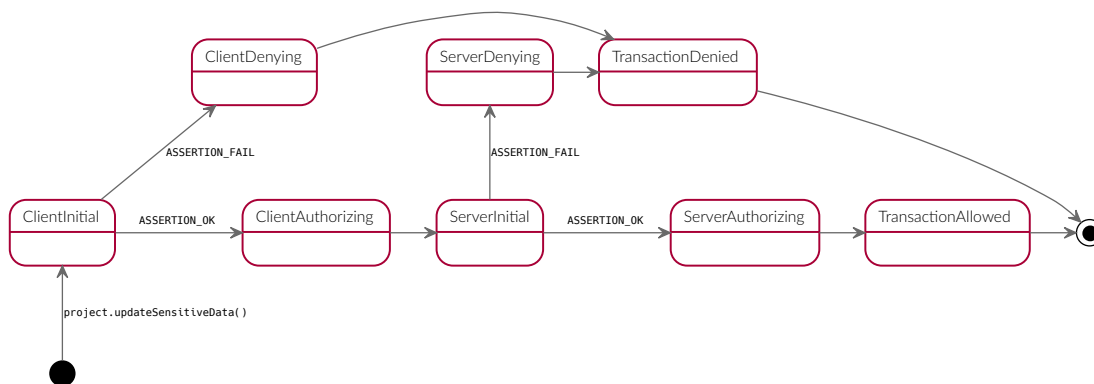


FIGURE 4.16 – Double contrôle d’accès client-serveur avec Astronomy et Serrurier

25. On pourra se référer à l’exemple `paranoid-reporter` donné dans la documentation de la librairie [62].

missions (Figure 1.2, *rel.* 5). Avec Serrurier, la résolution de portée (*scope*) peut être « automatique » en implémentant, pour une « classe » donnée, la fonction `getScope`²⁶. L'exemple est donné en Annexe E.3.3, ligne 60. Pour que la transaction réalisée par la méthode `updateUserRoleInProject` soit effectuée, l'utilisateur courant doit avoir la permission `project$updateUserRole` de portée `projects.XXX` où `XXX` correspond à l'identifiant unique du projet. Cette portée est renvoyée par la méthode `getScope` citée précédemment. Cette résolution de portée est non négociable, car elle garantit qu'un utilisateur n'ait pas d'accès à tous les projets dès lors qu'il accède à au moins l'un d'entre eux.

Mais nous proposons d'aller plus loin qu'une simple résolution de portée de l'objet à lui-même²⁷. En effet, on pourrait déclarer, de manière statique, quelle « classe » d'objets résout la portée de quelle autre. Nous proposons d'appeler ces classes qui délèguent la résolution de portée à une autre classe des « classes paresseuses », et les classes qui résolvent la portée pour d'autres classes des « classes déléguées » (Figure 4.17). De manière analogue, on parlera d'« objet paresseux » et d'« objet délégué » pour des instances de ces classes respectives. Pour qu'une telle délégation puisse fonctionner, une contrainte évidente est que l'objet paresseux doit avoir une et seulement une référence à l'objet délégué.

Dans le Tableau 4.8, on projette ce que pourrait donner la délégation appliquée aux classes de COLAB 0.5. Le champ transitif est le champ qui référence — avec un identifiant unique — l'objet délégué depuis l'objet paresseux. C'est lui qui va permettre de construire automatiquement la portée. On propose que la portée soit générée de la manière suivante : *NomDeClasseDéléguee+s+.+IdentifiantUnique*²⁸. On remarquera d'une part qu'une classe peut s'autodéléguer

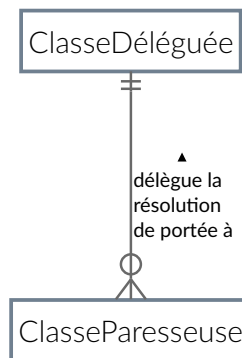


FIGURE 4.17 – Caption

26. Dans le code source de la librairie Serrurier, `partition` remplace `scope`. Ceci est dû à la librairie `alanning-roles`. Nous préférons utiliser « *scope* » qui nous semble plus intelligible.

27. Pour rappel, `getScope` renvoie une portée construite avec le motif suivant : `projects.XXX`.

28. `+` est un opérateur de concaténation.

TABLEAU 4.8 – Exemple de délégation avec des classes de COLAB 0.5

Classe paresseuse	Classe déléguée	Champ transitif
Project	Project	<i>_id</i>
Experiment	Project	<i>projectId</i>
AnnotatedTask	User	<i>ownerId</i>
User	User	<i>_id</i>

(Project, User). On l'appelle alors « classe autonome ».

Pour appliquer une gestion de rôle à une classe `Astronomy`, on pourrait utiliser un décorateur de classe. Les paramètres de ce dernier seraient une référence à la classe déléguée ainsi que le nom du champ transitif. On s'approche alors du paradigme de programmation par aspect. Les permissions nécessaires à l'exécution d'une méthode pourraient être générées automatiquement avec le motif suivant : *NomDeClasse*+.+*NomDeMethode*. On peut aussi considérer de la délégation multiple, avec certaines permissions explicitement déléguées à une autre classe que la classe déléguée de base. Enfin, on pourrait définir la « propriété » comme un cas particulier de la délégation où la classe déléguée est la classe « User ».

Son caractère déclaratif est un des points forts du modèle de rôles *Object Scope Delegate Role Based Access Control* (OSD-RBAC) que nous proposons. De plus, il peut permettre très facilement :

- d'implémenter une suspension de compte utilisateur. Pour cela, il suffit ou bien de retirer les permissions de l'utilisateur sur lui-même (via la portée *User._id*), ou bien d'implémenter une « *scope blacklist* » qui interdit pour un sujet donné la résolution de permissions sur la portée *User._id*. Cette dernière solution est préférable car non destructrice.
- d'implémenter un **Contrôle d'Accès Discretionnaire (CAD)**. Il suffit pour cela d'autoriser l'utilisateur à déléguer certaines de ses permissions. Plutôt que d'autoriser la délégation par défaut, ce qui risque de compromettre la sécurité de l'application, on pourrait définir un décorateur de méthode qui autorise la délégation pour des permissions une par une.
- de générer un diagramme de rôles et permissions à partir du code source.
- de créer de nouveaux rôles très rapidement.

Il faudrait aussi pouvoir définir des permissions déclaratives pour les opérations *Create Read Update Delete*.

Chapitre 5

Spécifications partielles

Ce chapitre présente une ébauche de spécifications d'interface graphique ([section 5.1](#)) ainsi que la description succincte du coopérativisme académique des plateformes ([section 5.2](#)).

5.1 CoLAB

Dans cette section, on propose d’offrir une base aux spécifications de l’application CoLAB. Dans un souci de lisibilité et pour ne pas influencer le *design*, les esquisses sont fournies en teintes de gris. Seuls les éléments dont les couleurs peuvent être définies par l’utilisateur apparaîtront dans des teintes qui n’appartiennent pas au gradient noir-blanc.

5.1.1 Plan de site

Dans le [Tableau 5.1](#), le plan de site de l’application est proposé. Trois remarques :

- les paramètres de route sont identifiés en étant précédés d’un double point « : » ;
- sauf si l’indication « accessible aux visiteurs » est précisée, les routes énumérées ne sont accessibles qu’aux [utilisateurs](#) authentifiés sur la plateforme ;
- on utilise l’expression « route fille » pour désigner toute route qui est composée de la première. Par exemple, `/projects/:prj/studies` est une route fille de `/projects/:prj`.

TABLEAU 5.1 – Plan de site pour COLAB

Route	Description
<code>/user</code>	Page de gestion du profil de l'utilisateur connecté.
<code>/user/login</code>	Page d'authentification. Accessible aux visiteurs.
<code>/user/register</code>	Page de création de compte. Accessible aux visiteurs.
<code>/users/:user</code>	Profil de l'utilisateur <code>user</code> .
<code>/dashboard</code>	Tableau de bord relatif à l'utilisateur connecté. Il peut y voir ses projets, ses tâches, ses notifications. Aussi, il a accès au magasin libre des universités, à l'assistant à la création d'OAC et à la liste de projets.
<code>/workshop</code>	Liste des outils en cours de création — à l'état de brouillon — pour l'utilisateur connecté. Les outils dont la propriété est collectivisée dans une étude sont aussi accessibles.
<code>/workshop/:ttype/:tool</code>	Assistant à la création et l'édition d'OAC (LO2A) pour l'outil <code>tool</code> de type <code>ttype</code> . Le type peut avoir pour valeur <code>grid</code> (grille d'évaluation), <code>task-model</code> (MTAC) ou bien <code>observation-type-hierarchy</code> (hiérarchie de catégories d'observation).
<code>/store</code>	Magasin libre des universités. Offre une liste des OAC publiés, ainsi qu'une fonction de recherche.
<code>/store/tools/:ttype/:tool</code>	Page unique pour consulter un outil <code>tool</code> de type <code>ttype</code> . Le type peut avoir pour valeur <code>grid</code> (grille d'évaluation), <code>task-model</code> (MTAC) ou bien <code>observation-type-hierarchy</code> (hiérarchie de catégories d'observation).

— suite à la page suivante —

Plan de site pour COLAB— *suite*

Route	Description
<code>/projects</code>	Liste les projets qui sont ouverts à l’inscription, ainsi qu’une fonction de recherche.
<code>/projects/:org/:prj</code>	Page d’accueil d’un projet <code>prj</code> géré par l’organisme <code>org</code> . Il faut être collaborateur pour y accéder ainsi qu’à toutes les routes filles.
<code>/projects/:org/:prj/threads/:thread</code>	Fil de discussion <code>thread</code> relié au projet .
<code>/projects/:org/:prj/exps</code>	Liste des expérimentations. Elles sont floutées .
<code>/projects/:org/:prj/exps/:exp</code>	Une expérimentation <code>exp</code> floutées et en lecture seule, c’est-à-dire sans possibilité d’éditer des annotations .
<code>/projects/:org/:prj/config</code>	Atelier de configuration pour projet (LO5B). Il faut être responsable de projet pour y accéder ainsi qu’à toutes les routes-filles.
<code>/projects/:org/:prj/config/collaborators</code>	Gestion des collaborateurs . Il faut être responsable de projet pour y accéder.
<code>/projects/:org/:prj/config/data</code>	Gestion des données d’expérimentation. Il faut être opérateur en domotique pour y accéder.
<code>/projects/:org/:prj/config/general</code>	Gestion générale du projet. Il faut être responsable de projet pour y accéder.
<code>/projects/:org/:prj/studies/:std</code>	Page d’accueil d’une étude <code>std</code> . Il faut être associé d’étude pour y accéder ainsi qu’à toutes les routes-filles.
<code>/projects/:org/:prj/studies/:std/threads/:thread</code>	Fil de discussion <code>thread</code> relié à l’étude.
<code>/projects/:org/:prj/studies/:std/exps</code>	Liste des expérimentations qui peuvent être étudiées.
<code>/projects/:org/:prj/studies/:std/exps/:exp</code>	Atelier d’ AC pour l’expérimentation <code>exp</code> (LO2B).

— suite à la page suivante —

Plan de site pour COLAB– *suite*

Route	Description
<code>/projects/:org/:prj/studies/:std/config</code>	Atelier de configuration pour étude (LO5C). Il faut être gestionnaire d'étude pour y accéder ainsi qu'à toutes les routes filles.
<code>/projects/:org/:prj/studies/:std/config/tools</code>	Espace de configuration des OACs pour l'étude.
<code>/projects/:org/:prj/studies/:std/config/coord</code>	Espace de coordination, avec l'établissement d'une feuille de route et de jalons.
<code>/projects/:org/:prj/studies/:std/config/associates</code>	Gestion des associés d'étude .
<code>/organizations/:org</code>	Page d'accueil de l' organisme <code>org</code> . À l'intérieur, on peut y voir les différents sites et projets gérés par l'organisme. La page est accessible à tous les utilisateurs
<code>/organizations/:org/config</code>	Atelier de configuration pour organisme (LO5A). Il faut être administrateur d'organisme pour y accéder.

5.1.2 Esquisses d'interface graphique

Les esquisses ¹ donnent un aperçu des principales fonctionnalités accessibles dans l'interface graphique relativement aux routes qui ont été jugées essentielles ². Elles sont visibles en [Annexe A](#). Ces esquisses n'ont pas de valeur contraignante et servent principalement à donner une forme plus complète à COLAB. Elles ne présentent pas de manière exhaustive toutes les fonctionnalités détaillées dans la liste des exigences ([section 3.7](#)). Elles ne supplantent pas non plus la création de maquettes par un *designer* et aucune recherche esthétique particulière n'a été mise en œuvre.

5.2 Coopérativisme académique des plateformes de recherches

Nous proposons ici l'ébauche d'un modèle de gouvernance et d'exploitation inspiré directement du coopérativisme des plateformes ([section 1.5](#)).

5.2.1 Les déficits invisibles du milieu académique

En 2015, l'économiste MAZZUCATO démystifie la croyance, bien ancrée dans l'imaginaire collectif, d'une dichotomie entre une recherche publique monolithique et stérile qui serait largement dépassée par le *leadership* d'un secteur privé innovant et audacieux [51]. Pour illustrer son propos, elle expose plusieurs études de cas célèbres de la Silicon Valley, à l'instar de Apple[®]. Toutes ces histoires ont en commun le « moissonnage » des fruits de la recherche publique par des acteurs privés, et la confiscation de bénéfices d'exploitation qui devraient, en toute logique, revenir à la société civile.

L'auteur d'un article du *Guardian*, daté de mai 2017, s'insurge contre l'usurpation de certaines entreprises technologiques qui, faute de savoir innover, font mine de le faire [69]. TARNOFF fait le constat amer que le capitalisme *tech* s'est approprié un mécanisme de rente inhibiteur d'innovation, relayant la critique faite par le mouvement du coopérativisme des plateformes.

Le lien avec le monde académique est direct : le modèle de financement de la recherche — au Canada aussi bien qu'aux États Unis et en France — est fondé sur les subventions, pourvues à condition qu'elles soient dépensées dans des délais impartis. Lorsque ces recherches ont conduit à la conception et la création de logiciels, ces derniers peuvent-être diffusés sous licence *Open*

1. *Wireframe* en anglais.

2. Faute de temps et de ressources, toutes les routes n'ont pas pu être esquissées.

Source. Mais le maintien de ces projets et l'engagement de frais de fonctionnement³ tient à un fil, souvent autour du bénévolat et de campagnes de don à l'instar du logiciel *v1c*. Les partenariats privés constituent l'essentiel de l'alternative aux subventions publiques. Mais dans ce scénario, il est très rare que les produits soient distribués sous licence *Open Source*.

En plus, les frais de maintenance d'une plateforme en ligne — structure nécessaire à la collaboration — sont plus élevés que pour des logiciels de bureau. Par conséquent, ces modes de financement ne permettent en général ni aux chercheurs, ni à la société civile, d'exploiter le fruit des sommes engagées en leur nom. Et c'est regrettable à plus d'un titre. D'une part, lorsque ces logiciels peuvent servir aux chercheurs, ces derniers n'ont pas toujours la capacité d'en assurer le déploiement, et sont amenés à dépenser leurs subventions dans des logiciels tiers payants. Il manque alors la structure pour maintenir et assurer les frais fonctionnels dudit logiciel. L'argent public sert alors à acheter des licences de logiciels de niche, souvent vendues à des prix prohibitifs se chiffrant en plusieurs dizaines de milliers de dollars. C'est ce qu'on pourrait appeler les « pertes d'entrée » du modèle actuel de financement de la recherche.

D'autre part, comme les subventions viennent de la société civile par l'intermédiaire de l'impôt, elles devraient y retourner sans être accaparées par des acteurs privés. C'est ce qu'on peut appeler les « pertes de sortie » du modèle actuel de financement de la recherche. Enfin, la mutualisation des moyens entre acteurs et pays devrait relever de l'évidence dans le milieu académique, alors que sa mission première est d'assurer la transmission et l'enrichissement des savoirs humains. Le manque à gagner de cette mutualisation, théorisé par le mouvement des communs, pourrait être désigné comme « les pertes de mutualisation ». Le coopérativisme académique des plateformes de recherches se présente alors comme une tentative pour réduire ces trois pertes liées au modèle actuel de financement de la recherche.

5.2.2 Retour aux communs

L'idée d'associer les communs au monde académique n'est pas nouvelle. L'*Open Science Framework* par exemple, offre un gestionnaire pour projets de recherche dans le monde académique⁴. Sur leur page d'accueil, il est inscrit « A scholarly commons to connect the entire research cycle. » Mais son mode de financement est assez classique puisqu'il vit principalement de dons et de subventions par de grands groupes privés. Une telle coopérative pourrait au départ être financée par des fonds publics, mais à terme ce seraient ses utilisateurs qui en détiendraient

3. On pense notamment à la mise en place de serveurs ou la location d'un nom de domaine.

4. Ce produit est très différent de COLAB. Il correspond à un gestionnaire de projet classique, agrémenté de fonctionnalités pertinentes pour la recherche.

des parts (modèle mutualiste). Cette structure serait alors relativement indépendante des secteurs publics et privés, à la manière du modèle des communs. La coopérative aurait alors une mission d'intérêt général mandatée par la société civile. La perspective d'augmenter l'autonomie financière des laboratoires participants inscrirait le coopérativisme académique des plateformes de recherche dans la lignée du modèle Humboldtien où la liberté académique est totale et inconditionnelle.

L'objectif d'une telle coopérative serait donc de favoriser le développement d'outils d'assistance à la recherche. Parmi les utilisateurs, on peut penser aux **organismes** de recherche. Pour développer un modèle économique pérenne, une idée consisterait à créer deux licences d'utilisation bien distinctes. L'une, la « licence mutualiste » est destinée aux organismes non lucratifs, dont les organismes de recherche et d'enseignement. Cette licence avec abonnement mensuel offrirait des droits à l'**organisme** vis-à-vis de la coopérative. Par exemple :

- après un an de « cotisation », droit de participer à la gouvernance (gain de voix) ;
- après un an de « cotisation », droit de recevoir avec les autres membres mutualistes une partie des bénéfices de la coopérative, la part de bénéfices étant décidée par le conseil (droit aux bénéfices). La seule condition est que ces bénéfices doivent financer des recherches valides suivant la charte de la coopérative.

La deuxième, la « licence privée », s'adresserait aux organismes privés et aux entreprises. C'est une licence qui ne pose aucune condition sur l'utilisation du logiciel. Son utilisation peut amener, indirectement, à générer du bénéfice, sans que le groupe soit inquiet. Mais elle pourrait coûter plus cher, et n'offre aucun droit mutualiste.

Une telle coopérative, formée d'au moins une dizaine à une quinzaine de salariés permanents, serait aussi une place de choix pour les collaborations avec des laboratoires de recherche. Des projets de création logicielle pourraient être réalisés de façon bilatérales, ainsi que des stages voire des maîtrises et doctorats de sciences appliquées.

Conclusion

COLAB est encore, en l'état, un projet dans sa phase initiale. Le travail réalisé est un premier défrichage, une première exploration du problème et des perspectives qu'une telle solution pourrait offrir. Grâce aux différents prototypes élaborés au cours de cette étude, et grâce au questionnaire d'utilisabilité proposé aux premiers utilisateurs de l'outil, nous avons pu observer qu'il suscitait un intérêt manifeste. L'outil a le potentiel de bousculer la collaboration entre chercheurs en analyse comportementale, et nous allons développer dans les lignes qui suivent les indices qui nous amènent à le croire.

La caractéristique de **réflexivité (C.III)** est un élément central de l'application. Voici un rappel de la définition :

« *Épistémologie.* La réflexivité en sciences, c'est la capacité de regard et d'introspection sur sa propre démarche scientifique, voire l'étude de sa propre méthode. Le concept a été introduit par Pierre BOUDIEU. Il a notamment étudié les mécanismes sociaux à l'oeuvre dans la "sélection naturelle" appliquée aux théories scientifiques. »

L'application a la capacité de stimuler la réflexivité de plusieurs manières. D'abord, par la transparence des évaluations, avec la possibilité pour un **visiteur d'étude** — un chercheur extérieur au projet — d'inspecter les **annotations** qui ont abouti à des résultats publiés. De cette manière, il peut « retracer intégralement l'évaluation d'un participant par un évaluateur » et laisser apparaître « toutes les observations intermédiaires qui ont conduit aux conclusions de l'évaluateur ». Cette caractéristique permet la traçabilité des résultats de recherche, qui peut aussi avoir un grand intérêt pédagogique. Ensuite, la facilité offerte pour les **associés d'étude** de se référer à des événements précis d'une **expérimentation** et d'en débattre dans un espace dédié va inciter les chercheurs à discuter de leur compréhension des critères d'évaluation d'une part, du comportement des sujets et donc de leur objet d'étude d'autre part. Enfin, la possibilité pour un chercheur de faire des **évaluations réflexives** et donc de définir la confiance qu'il accorde à une annotation qu'il a lui-même produite ouvre la voie à de nouvelles discussions épistémologiques. Nous

pensons que tous ces éléments de réflexivité aident à rendre plus limpide la méthode de l'analyste, et par conséquent ils pourraient constituer un vecteur important d'interdisciplinarité (O.I). D'autres mécanismes comme l'évaluation interjuge participent à renforcer cet aspect. L'interdisciplinarité est aussi facilitée par les espaces de discussion et les différents mécanismes de collaboration proposés dans l'application. On peut distinguer deux axes de collaboration :

1. l'axe « chercheur-chercheur », prépondérant, encouragé par l'espace de discussion dédié à une *étude* ;
2. l'axe « chercheur-opérateur », incité par l'espace de discussion dédié au *projet*.

Cette collaboration ne serait pas possible sans la caractéristique d'accessibilité (C.I), vérifiée par l'utilisation d'une *AWM* dans une architecture semi-centralisée. Aussi, la caractéristique de confidentialité est garantie avec le système de contrôle d'accès *Object Scope Delegate Role Based Access Control* (OSD-RBAC) en tandem avec les exigences de sécurité qui contraindront les versions exploitables de COLAB. Enfin, un modèle modulaire permettra à la communauté d'implémenter de nouveaux *outils d'analyse*, cette modularité étant la solution envisagée pour satisfaire l'objectif de généricité (O.II), qui a son tour renforcera le potentiel d'interdisciplinarité (O.I).

COLAB a aussi le potentiel de se distinguer des logiciels concurrents. Tandis qu'*Observer XT* et *Captiv* sont des outils très riches, « usine à gaz », COLAB est relativement simple d'utilisation. La prise en compte des contraintes liées à une évaluation à l'aveugle avec la mise en place du *floutage d'expérimentation* différencie COLAB de ses concurrents. Ce floutage est permis par la séparation entre phase d'acquisition et d'analyse. Aussi, l'API SCÉNARII facilite le découplage entre les dispositifs d'acquisition et le logiciel d'analyse ce qui le rend beaucoup plus générique et indépendant de technologies propriétaires (adaptabilité, O.III). Il est orienté « analyse comportementale en environnement “intelligent” », il est collaboratif, *réflexif* et en ligne. Il est *Open Source* (C.IV) et enfin il repose sur un modèle d'exploitation sous forme d'une plateforme coopérative (O.IV). Le coopérativisme académique des plateformes de recherche serait alors un moyen envisageable pour rendre les organismes de recherche plus autonomes, rapprochant le monde académique du modèle Humboldtien.

De nombreux travaux restent encore à faire. Notamment, la phase d'analyse connaîtra de nouveaux cycles avant qu'une stabilisation suffisante permette une spécification complète. Il faudra par exemple raffiner la description des éléments suivants :

- l’interface API SCÉNARII ;
- les étapes qui permettent d’importer un ensemble d’expérimentations depuis une API SCÉNARII vers un *projet* avec des mécanismes de synchronisation manuels et automatiques ;
- les systèmes transformationnels, soit les fonctions d’exploitation ainsi que le module d’export ;
- les outils d’analyse existants, comme les *MTACs* et les *grilles d’évaluation* ;
- le modèle modulaire pour outils d’analyse ;
- le modèle du coopérativisme académique des plateformes de recherche ;
- une définition formelle de l’OSD-RBAC ;
- et certainement préciser de nombreux aspects qui n’ont pas pu être anticipés à ce stade.

Qui plus est, si le projet se développe suffisamment, l’application pourra bénéficier de nouvelles fonctionnalités et intégrations, démultipliant encore son potentiel collaboratif :

- intégration à l’*Open Science Framework*, logiciel académique pour la gestion de projet ;
- intégration à *Dataverse*, logiciel académique pour publication de données ;
- intégration à des messageries professionnelles, comme *Slack* et *Rocket Chat* ;
- support des vidéos panoramiques dans l’API SCÉNARII, pour améliorer le confort des *analystes* ;
- support des listes de contrôle pour *conditions initiales d’expérimentation*, qui seront utiles aux *opérateurs en domotique* et *opérateurs d’expérimentation*.

Nous espérons alors qu’un jour, COLAB apportera une modeste contribution à l’édifice des technologies réflexives qui aident femmes et hommes à mieux comprendre leur mécanique intérieure, à l’œuvre en eux-mêmes et chez les autres, pour la science et pour la société.

Annexe A

Esquisses légendées

Ces esquisses légendées accompagnent le **chapitre 5** de spécifications. Elles sont données au format paysage pour le confort de lecture.

A.1 Page de projet

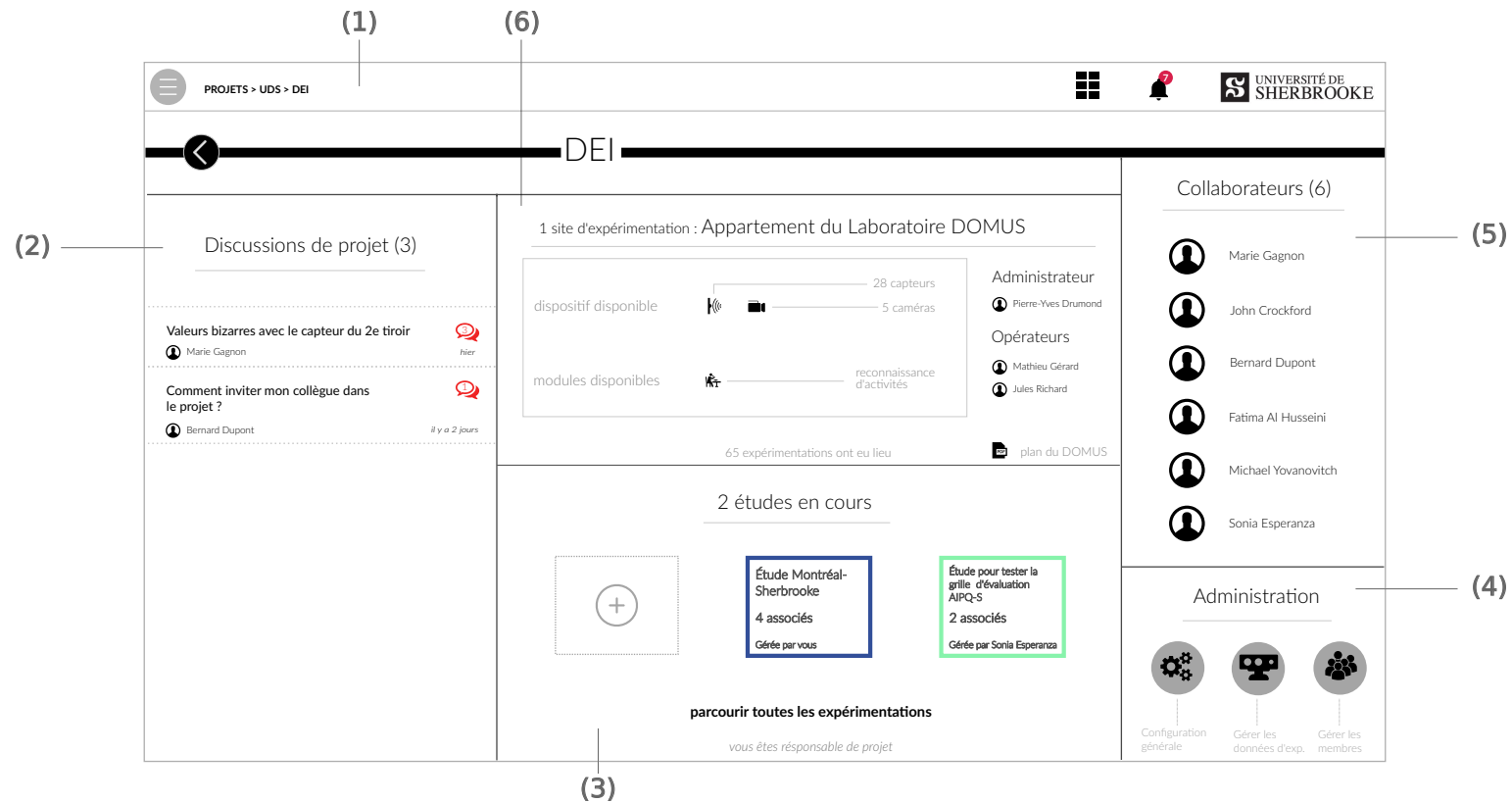


FIGURE A.1 – Esquisse de page pour projet /projects/uds/dei

(1) Barre de navigation. (2) Espace de discussion pour projet. En rouge, des sujets avec de nouvelles réponses parmi ceux où l'utilisateur connecté a participé. (3) Études liées au projet. Un clic sur l'une d'entre elles redirige vers la page d'accueil de l'étude (Figure A.2). Il est aussi possible de parcourir toutes les expérimentations relatives au projet. (4) Volet d'administration du projet, visible seulement pour les responsables de projet. (5) Liste des collaborateurs du projet. Un clic sur l'un d'entre eux redirige sur leur profil. (6) Site(s) d'expérimentation identifiés pour le projet. Les « dispositifs disponibles » désignent les appareils participant à l'acquisition de données. Les « modules disponibles » désignent des fonctionnalités optionnelles de l'API SCÉNARI, comme la reconnaissance d'activités et la détection de mouvement pour vidéo. Aussi, un fichier PDF du plan du site d'expérimentation ainsi que d'autres ressources peuvent être disponibles.

A.2 Page d'étude

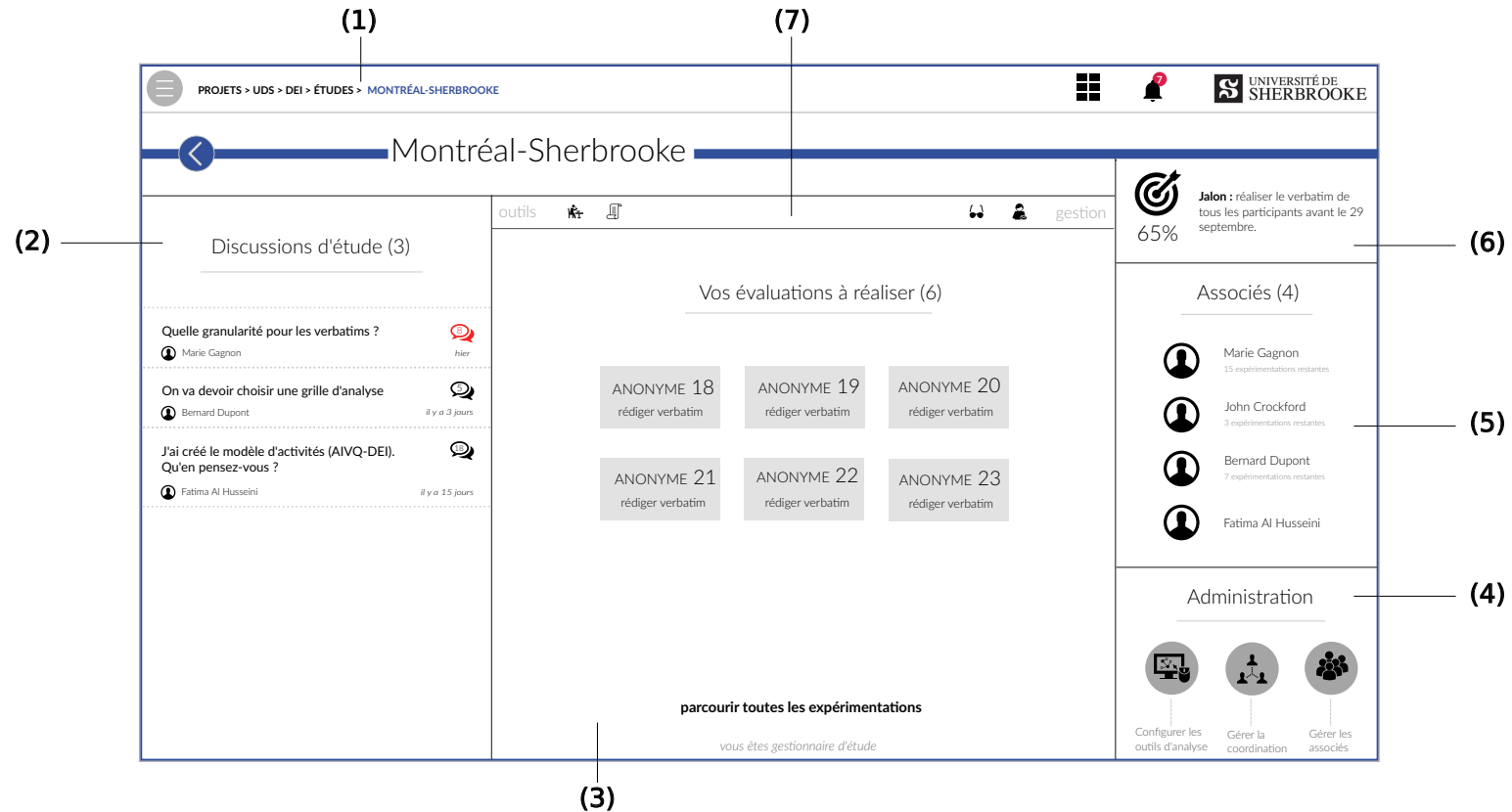


FIGURE A.2 – Esquisse de page pour étude /projects/uds/dei/studies/montréal-sherbrooke

(1) Barre de navigation. (2) Espace de discussion pour étude. (3) Liste des évaluations à réaliser par l'associé d'étude connecté, relativement au jalon visé par le groupe d'étude. Il peut aussi parcourir toutes les expérimentations reliées à l'étude. (4) Administration du projet, visible seulement aux gestionnaires d'étude. (5) Liste des associés d'étude. (6) Jalon fixé par un associé d'étude, et son taux de clôture. (7) Informations sur l'étude courante. Chaque icône peut être survolée pour obtenir une infobulle. À gauche, les outils déployés (respectivement, un MTAC et un verbatim). À droite, les règles de gestion collaborative (respectivement, floutage activé pour évaluation à l'aveugle et nouveaux associés sur invitation).

A.3 Page de configuration des outils pour étude

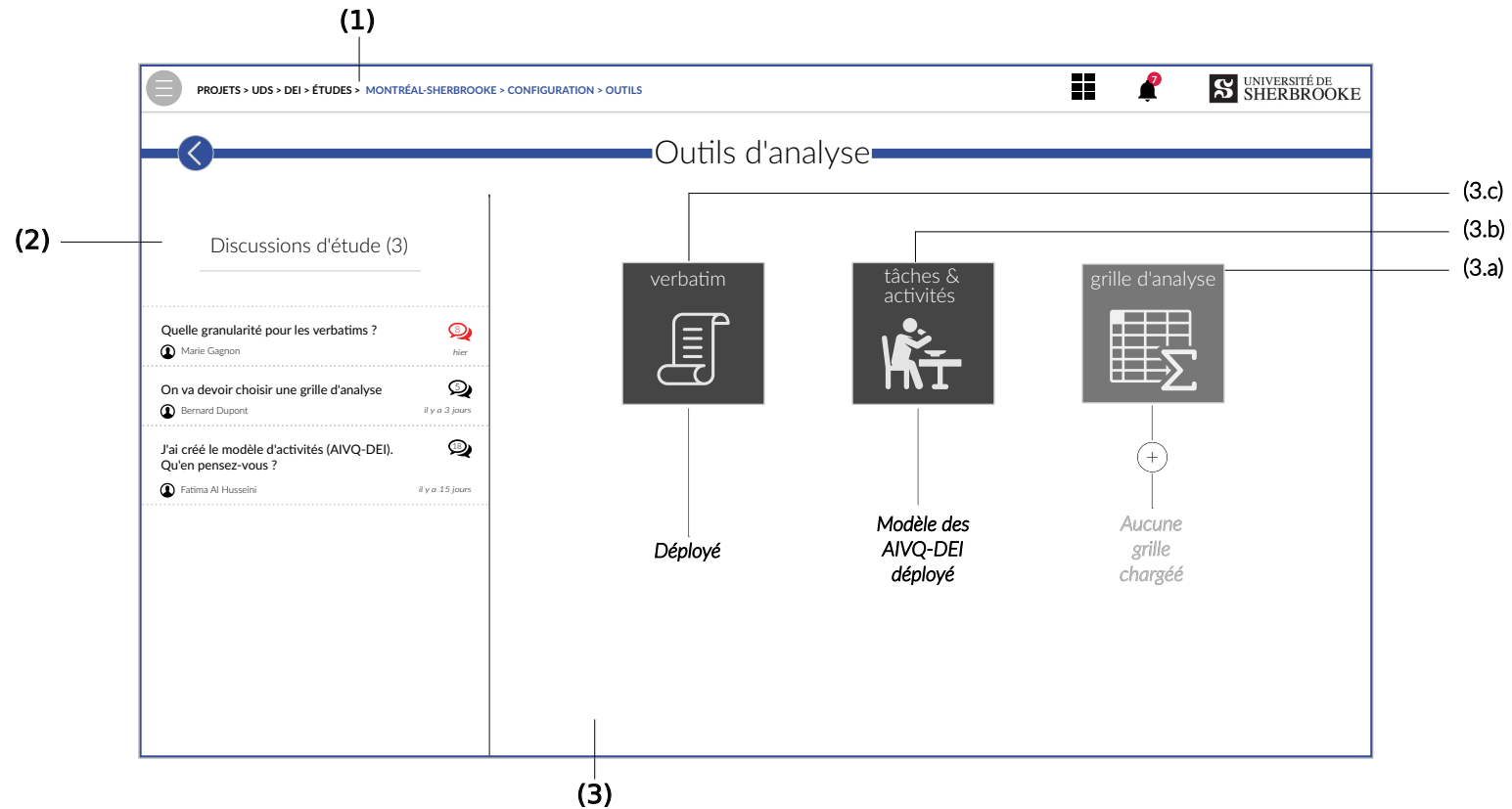


FIGURE A.3 – Esquisse de page de configuration d'outils pour étude /projects/uds/dei/studies/montréal-sherbrooke /configuration/outils

Cet page n'est accessible qu'à un **gestionnaire d'étude**. (1) Barre de navigation. (2) Espace de discussion pour étude. (3) Espace de configuration. (3.a) Configuration de **grille d'évaluation** déployé pour l'étude. En l'occurrence, aucune grille n'est chargée. Le **gestionnaire d'étude** peut appuyer sur le « + » pour déployer une grille. Il aura alors le choix entre créer une nouvelle grille et charger une grille existante, parmi son inventaire ou bien en se rendant dans le **magasin libre des universités**. (3.b) Configuration du **modèle de tâches** déployé pour l'étude. Le déploiement se fait de manière analogue à la grille. (3.c) Configuration du module verbatim. Le déploiement se fait via un interrupteur (*on/off*).

A.4 Atelier d'analyse comportementale pour expérimentation

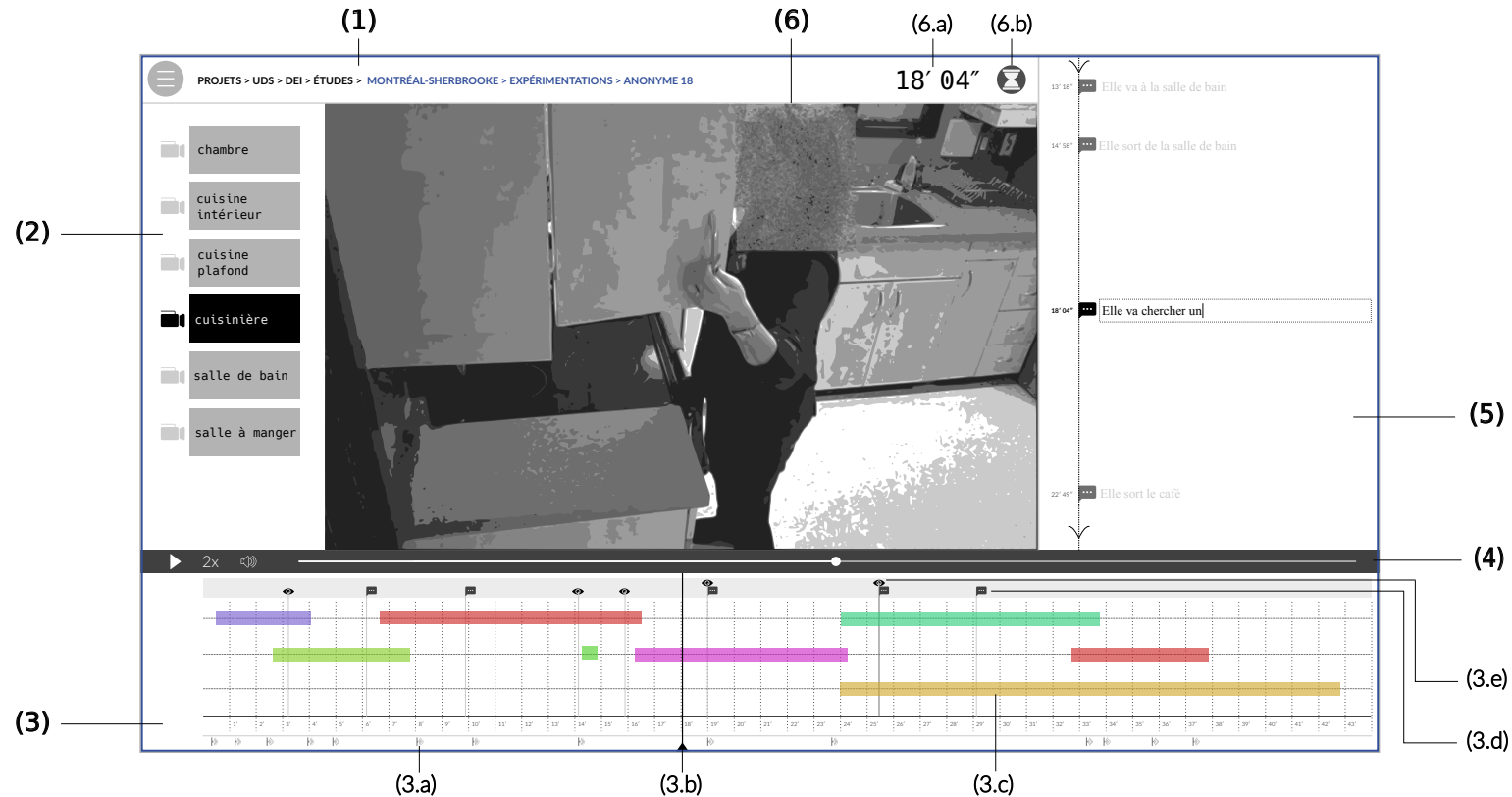


FIGURE A.4 – Esquisse d'Atelier d'AC /projects/uds/dei/studies/montréal-sherbrooke/exps/anonymus18

On peut observer un cadre bleuté autour de cette page. Cette couleur correspond à la couleur de l'étude courante, et elle est aussi présente dans le fil d'Ariane. (1) Barre de navigation. (2) Sélecteur de sources. (3) Explorateur temporel. Un survol avec le curseur de souris laisse apparaître un curseur de temps fantôme avec le temps relatif exact. (3.a) Événements de capteurs. (3.b) Curseur de temps. (3.c) Tâche réalisée par le participant. Visibles seulement si un modèle de tâches est déployé. La couleur correspond au type de tâche, déterminée par le modèle de tâches. (3.d) Observation de verbatim. Visibles seulement si le module verbatim est déployé. (3.e) Observation de grille. Visibles seulement si une grille d'évaluation est déployée. (4) Barre de contrôles. (5) Trame temporelle pour verbatim. Visible seulement si le module verbatim est déployé. (6) Lecteur vidéo embarqué. (6.a) Temps, relatif ou absolu. (6.b) Interrupteur de temps relatif / temps absolu.

A.5 Structure générale

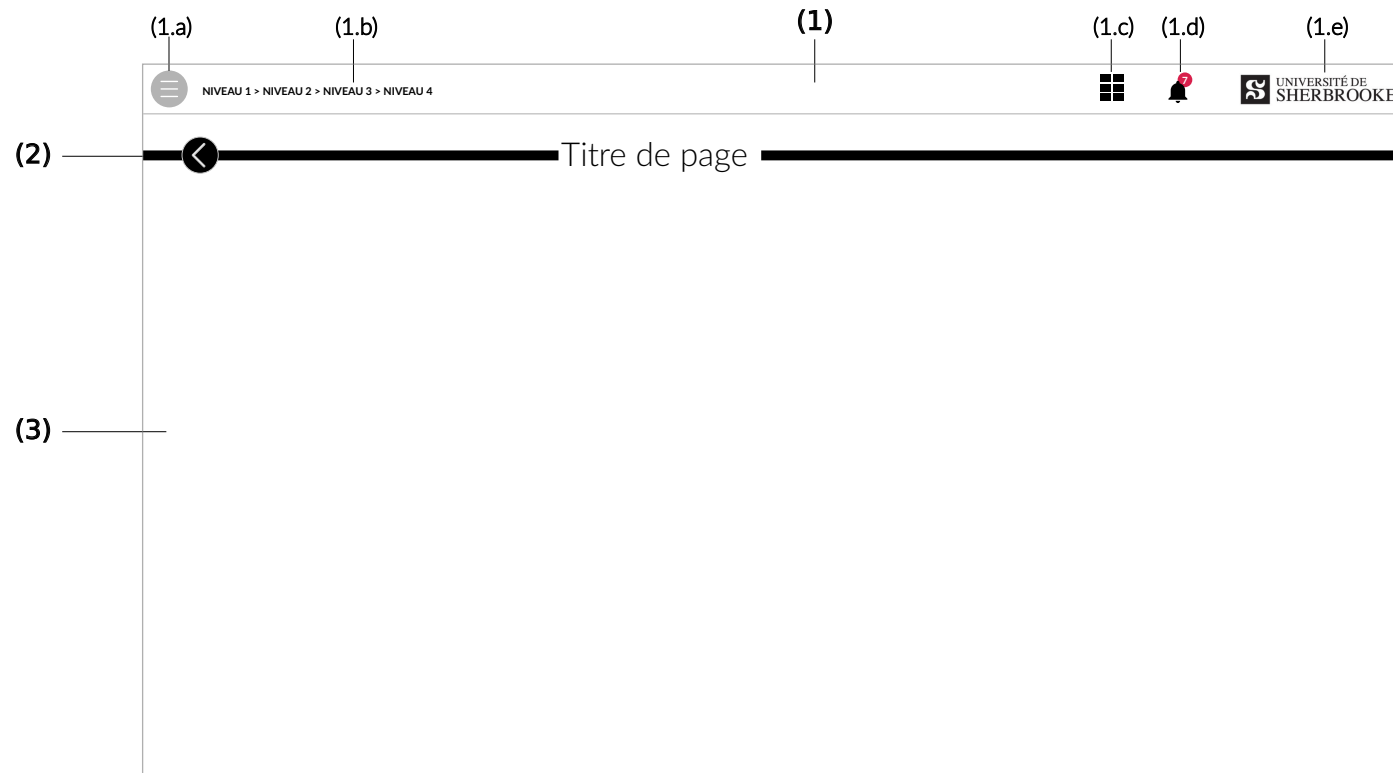


FIGURE A.5 – Structure générale d'une page de l'application COLAB

(1) Barre de navigation. (1.a) Bouton pour déplier les accès rapides (Figure A.6). (1.b) Fil d'Ariane. Un clic sur un niveau redirige vers l'URL correspondant à ce niveau. (1.c) Lien vers le tableau de bord. (1.d) Bouton pour notifications (Figure A.7). (1.e) Logo de l'organisme d'affiliation. Redirection vers la page d'accueil de l'organisme sur clic. (2) [Optionnel] Barre de titre. Le bouton « < » permet de reculer d'une page dans l'historique. (3) Corps de page.

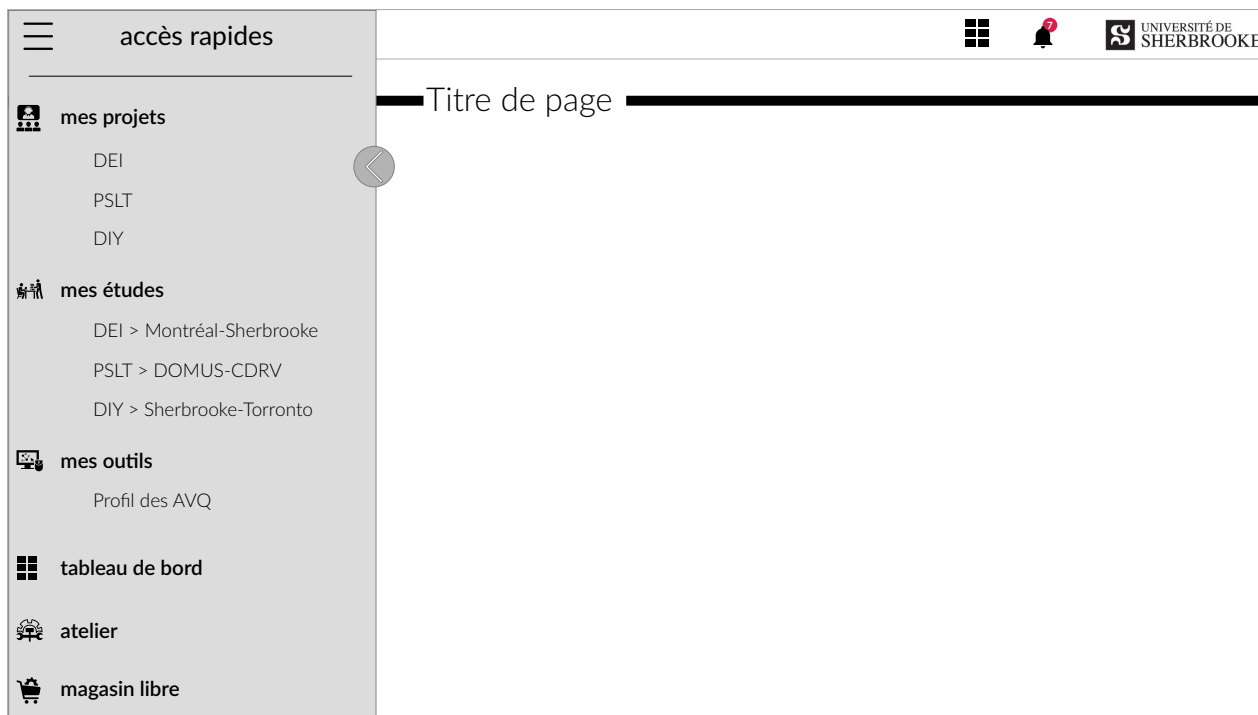


FIGURE A.6 – Accès rapides dans l'application COLAB

Les accès rapides permettent en deux clics de bifurquer sur un **projet**, une **étude** ou bien de charger un **OAC** dans l'atelier. Le tableau de bord et le **magasin libre des universités** sont aussi accessibles.



FIGURE A.7 – Notifications dans l'application COLAB

Il existe trois types de notifications. Les notifications sociales, liées à des discussions entre **collaborateurs** ou **associés d'étude**, les demandes d'accès liées à des **responsabilités** et les notifications de collaboration, relatives à des modifications d'annotation, d'outils... etc. Parmi les premières, on notera d'une part la possibilité de mentionner quelqu'un dans un fil de discussion et d'être alors notifié, d'autre part la possibilité pour un **associé d'étude** d'attirer l'attention d'autres associés sur un moment précis d'une **expérimentation**. Différentes interactions sont prévues pour chaque type de notification. Lorsque l'**utilisateur** choisit l'action « reporter », la notification disparaît du fil de notifications et devient une tâche à réaliser accessible dans le tableau de bord.

Annexe B

Notations

B.1 UML

La notation UML 2.0 est utilisée. Pour faire référence à des ressources externes, le symbole \Rightarrow est utilisé.

B.2 Diagrammes logiques de données « Crow's Foot »

La notation « Crow's Foot » utilisée est celle de James MARTIN. C'est une notation similaire aux diagrammes entité-relation, mais la cardinalité est exprimée par des symboles. Dans la [Figure B.1](#), la cardinalité des relations entre entités est représentée, ainsi que la correspondance avec les diagrammes entité-relation.

Un exemple simplifié est proposé en ([Figure B.2](#)). Les diagrammes logiques de données simplifiés omettent les colonnes (ou propriétés) des tables (ou entités) généralement fournies. Le diagramme se lit ainsi :

- « A » est composé de zéro à plusieurs « B » ;
- « B » est relié à exactement un « A ».

—	Un et seulement un	[1]
> —	Un à plusieurs	[1 .. *]
+○—	Zero ou un	[0 .. 1]
>○—	Zero à plusieurs	[0 .. *]

FIGURE B.1 – Légende de cardinalité dans les diagrammes logiques de données « Crow's Foot »

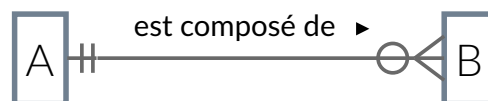


FIGURE B.2 – Diagramme logique de données « Crow's Foot », exemple simplifié

B.3 *Problem frames*

La Figure B.3 décrit notre syntaxe propre aux diagrammes *problem frames*, très largement inspirée par les notations de JACKSON BRAY . On se référera à l'Annexe C pour consulter la description complète des *problem frames*.

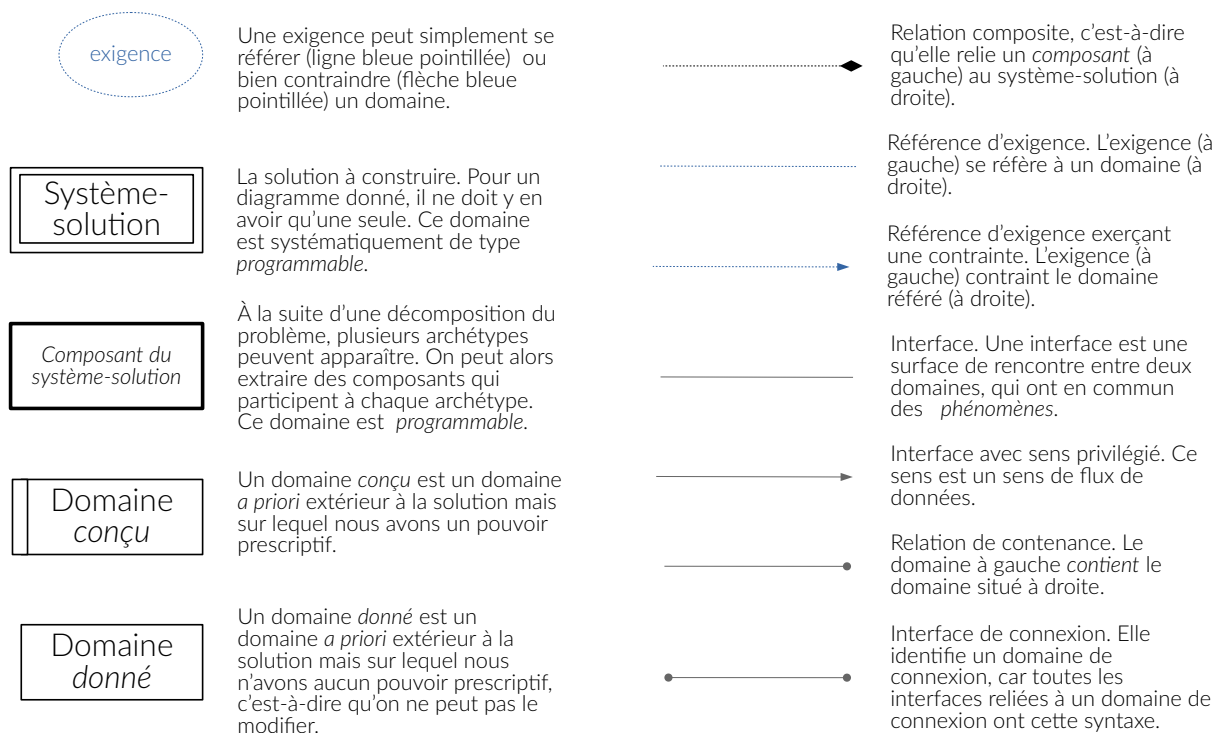


FIGURE B.3 – Légende des diagrammes *problem frames*

Annexe C

Problem frames ou problèmes-archétype

Les *problem frames* sont des archétypes de problèmes rencontrés dans l'ingénierie logicielle¹. Ils décomposent le **domaine du problème** en sous-domaines suivant une notation détaillée dans la **section B.3**. Ces sous-domaines peuvent être catégorisés suivant leurs propriétés (**section C.3**). Enfin, les archétypes connus sont décrits en **section C.2**. Dans les diagrammes qui vont suivre, les catégories (**Tableau C.1**) de sous-domaines compatibles avec un archétype sont indiquées entre accolades { T1 | T2 }, le symbole « | » désignant l'opérateur logique « ou ».

1. Nous proposons comme traduction française « problèmes-archétype » ou plus succinctement « archétypes ».

TABLEAU C.1 – Abréviations des catégories de domaine

Type	Abréviation
Statique	STQ
Dynamique	DYN
Inerte	INR
Actif	ACT
Spontané	SPN
Réactif	RCT
Programmable	PRG
Autonome	AUT
Docile	DOC

C.1 Introduction à la notation des problèmes-archétype

On se réfèrera à l’[Annexe B.3](#) pour consulter la notation complète. Les problèmes-archétype permettent, d’une part, d’identifier visuellement les domaines à considérer pour étudier le ou les problèmes à résoudre, et d’autre part de caractériser leur relation avec les exigences à produire.

Dans la [Figure C.1](#), un problème-archétype de type « atelier » (*confer section C.2*) est représenté. Plus précisément, il s’agit d’un logiciel de dessin vectoriel, similaire à Inkscape. Chaque rectangle de ce *problem frame* représente un sous-domaine.

Au centre, le logiciel et la machine qui l’héberge — c’est-à-dire le système-solution — sont symbolisés par un rectangle à bordure double². À droite, un « dessin » est un objet intangible contenu dans le système-solution. Cette relation d’inclusion peut être exprimée pour n’importe quel sous-domaine, et permet d’enrichir par itérations successives la description du problème. Les lignes joignant deux sous-domaines indiquent que ces domaines partagent des *phénomènes*³.

Évidemment, cet exemple est relativement simple, mais l’intérêt de cette méthode est de

2. C’est la notation privilégiée par BRAY.

3. La notion de « phénomène » est relativement abstraite et libre d’interprétation. JACKSON distingue des phénomènes *causaux* et *symboliques*. Les premiers s’illustrent, par exemple, par des événements ou des relations entre événements communs à deux domaines. Les seconds sont dits *symboliques* parce qu’ils se réfèrent à des *valeurs*, typiquement du texte ou des nombres. [40, page 83].

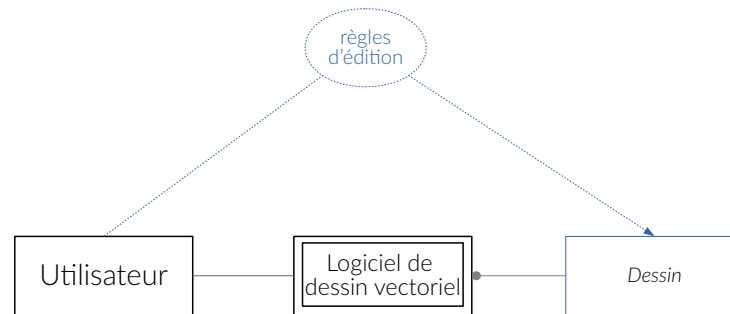


FIGURE C.1 – Problème-archétype élémentaire, un logiciel de dessin vectoriel

décomposer des problèmes complexes en une somme de problèmes élémentaires. Par exemple, on peut imaginer que ce logiciel soit pourvu d'un module pour exporter un dessin en PDF, ou encore un autre module pour importer une image dans le dessin. On propose dans la [Figure C.2](#) un *problem frame* composite, avec une composante « atelier », qui est la composante dominante (rectangle à bordure double), puis une composante « transformationnelle », identifiée par le rectangle à bordure grasse « Module d'exportation PDF ». Le losange⁴ dénote une relation de composition entre le système-solution et le module d'exportation PDF. Enfin, les éventuelles

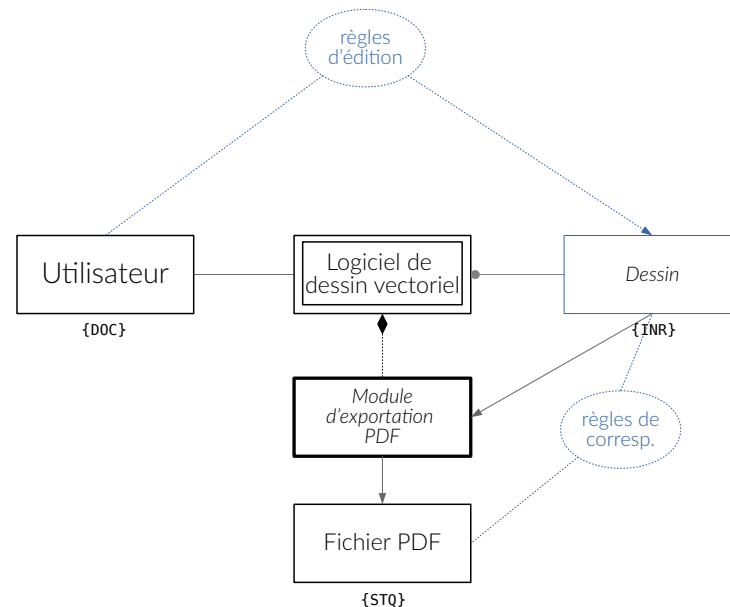


FIGURE C.2 – Problème-archétype composite, un logiciel de dessin vectoriel

4. Ni BRAY ni JACKSON ne proposent cette notation, mais nous l'avons trouvé particulièrement commode pour décrire des problèmes composites. Les deux auteurs susmentionnés proposent de dupliquer le système-solution en chacune de ses composantes (composante système d'information, composante transformation par exemple [7, page 112]).

flèches reliant deux sous-domaines symbolisent un sens privilégié de flux de données.

C.2 Typologie des problèmes-archétype

BRAY distingue cinq types de *frames* [7, page 94] :

1. système-atelier ;
2. système-contrôleur ;
3. système d'information ;
4. système transformationnel ;
5. système-connecteur.

Chacun de ces archétypes contraint les caractéristiques (section C.3) que peuvent avoir les sous-domaines qui en font partie. Ces catégories sont données entre accolades, par leur abréviation (Tableau C.1).

C.2.1 Système-atelier

L'archétype « système-atelier » s'applique lorsque le système doit effectuer des opérations sur un objet qui n'existe qu'à l'intérieur du système ; on appelle cet objet la « pièce de travail » (*workpiece*), et on parle alors de « domaine réalisé » (Figure C.3). Typiquement, on peut penser à de nombreux logiciels dont les artefacts créés ou modifiés n'existent qu'à l'intérieur du système, comme un éditeur de pistes audio, un éditeur de texte riche, un logiciel de dessin.

C.2.2 Système-contrôleur

Cet archétype s'applique lorsque le système contrôle le comportement d'une portion du **domaine du problème**. Le domaine contrôlé doit pouvoir réagir à des stimulus extérieurs d'une manière au moins partiellement prédictible, il peut donc être « réactif », « programmable » ou « docile » mais pas nécessairement tangible. Des règles comportementales définissent le comportement désiré du domaine contrôlé, et le système-contrôleur est en charge de s'assurer qu'elles s'appliquent.

Deux variantes existent : la variante automatique (*required behaviour frame*, Figure C.4) se distingue de la variante commandée (*commanded*, Figure C.5) par la manière que sont déterminées les règles comportementales. Dans la première, elles sont connues d'avance du contrôleur. Dans la seconde, elles sont déterminées par un opérateur. Parmi des exemples concrets, on peut citer un contrôleur pour ascenseur, un système embarqué de gestion du moteur dans un véhicule moderne, un système d'alerte-incendie.

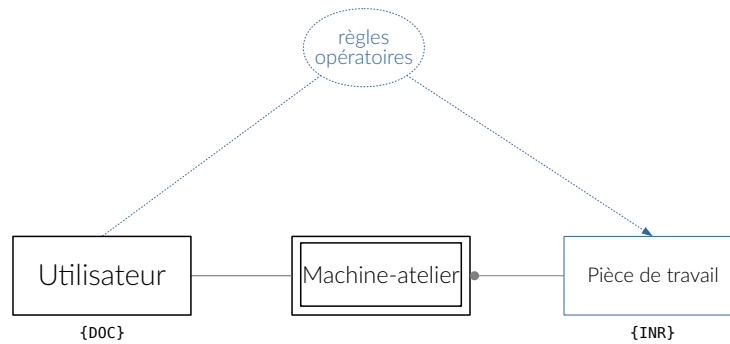


FIGURE C.3 – Problème-archétype : système-atelier



FIGURE C.4 – Problème-archétype : système-contrôleur, variante automatique

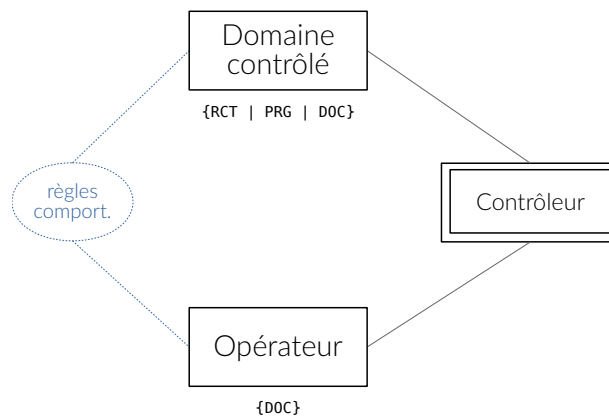


FIGURE C.5 – Problème-archétype : système-contrôleur, variante commandée

C.2.3 Système d'information

L'archétype « système d'information » est probablement le plus commun. Il est pertinent lorsque le système doit fournir des informations sur un domaine statique ou autonome, le « domaine observable »⁵. Deux variantes existent. Une première variante fournit une information continue sur le domaine observable (Figure C.6).

5. Les auteurs parlent du « Monde réel » tout en admettant qu'il puisse s'agir d'un domaine intangible, comme par exemple le domaine observé par un moniteur de charge système dans un système d'exploitation. On préfère « Domaine observable » qui est plus générique tout en renvoyant explicitement à l'objet observé du système d'information.

Un exemple typique serait le tableau de bord d’une voiture, avec des informations temps-réel sur le kilométrage, la vitesse, le niveau d’essence, et d’éventuelles défaillances. Dans la deuxième variante (Figure C.7), le système d’information ne livre pas l’information en continu, mais réagit à des requêtes pour fournir des rapports. Dans cette variante, deux types d’entrées distincts coexistent. Des requêtes d’une part, certainement commandées par un utilisateur ou une autre machine, et des actualisations du domaine observable d’autre part. Ces actualisations peuvent être éventuellement explicitées par l’ajout d’un domaine (Figure C.8).

On propose une simplification syntaxique du système d’information avec requêtes en (Figure C.9). Le « sélecteur » est un court texte décrivant quelles données sont requises, et la « réponse », optionnelle, décrit brièvement avec abstraction les valeurs de retour. Le couple sélecteur-réponse peut être référencé comme le serait le domaine « réponse » dans l’écriture non simplifiée. Une variante avec \uparrow et \downarrow peut être rencontrée lorsque le domaine « Requête » est situé en dessous du domaine « Système d’information ».

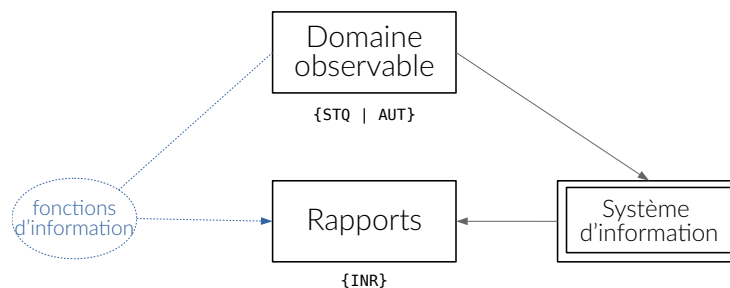


FIGURE C.6 – Problème-archétype : système d’information, variante temps-réel

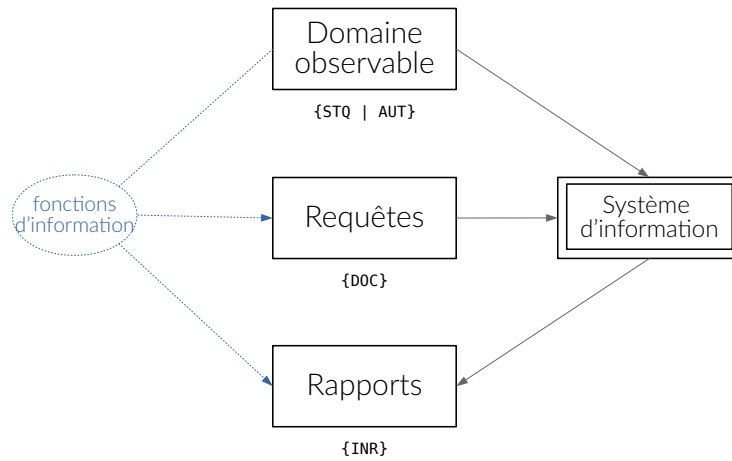


FIGURE C.7 – Problème-archétype : système d'information, variante sur demande

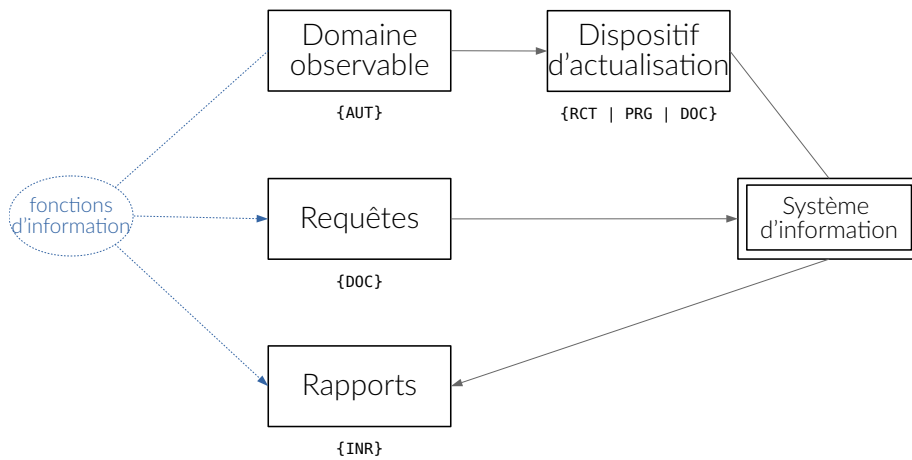


FIGURE C.8 – Problème-archétype : système d'information, variante sur demande avec dispositif d'actualisation

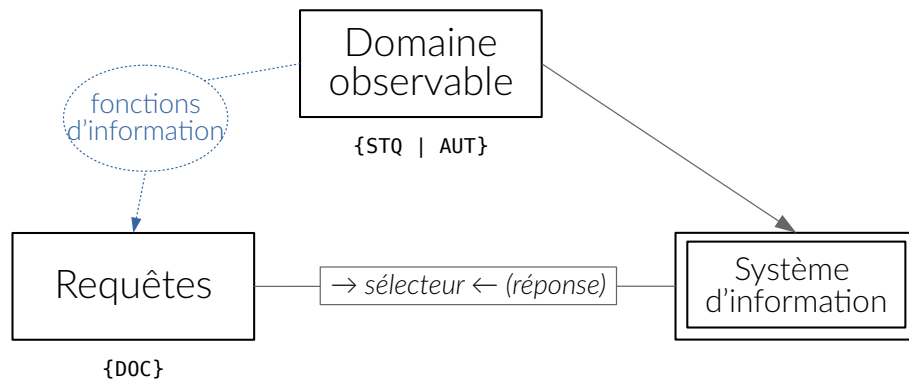


FIGURE C.9 – Problème-archétype : système d'information, variante sur demande simplifiée

C.2.4 Système transformationnel

L'archétype transformationnel est très simple. Il transforme des données entrantes, statiques, en données sortantes, inertes. Ce problème relève presque de la description d'une fonction, et les exigences consistent à faire correspondre des données structurées entrantes en des données structurées sortantes.

C.2.5 Système-connecteur

Cet archétype s'emploie lorsque le système maintient une correspondance entre sous-domaines qui ne sont pas directement reliés. Un système-connecteur typique serait, par exemple, un ensemble de capteurs qui permettent à un système d'information de traduire des phénomènes du monde observable (Figure C.11) en phénomènes qui lui sont intelligibles. Dans cette variante excentrée, le domaine intermédiaire est imposé (« appareil de connexion », ou ensemble de capteurs dans notre exemple), et le « système-connecteur » à concevoir exploite ou corrige les informations qui lui parviennent.

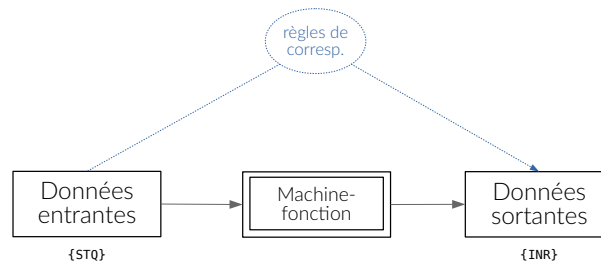


FIGURE C.10 – Problème-archétype : système transformationnel

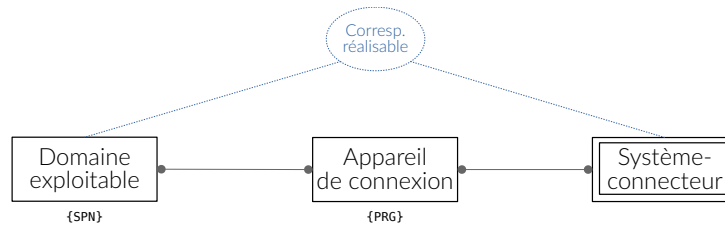


FIGURE C.11 – Problème-archétype : système-connecteur, variante excentrée

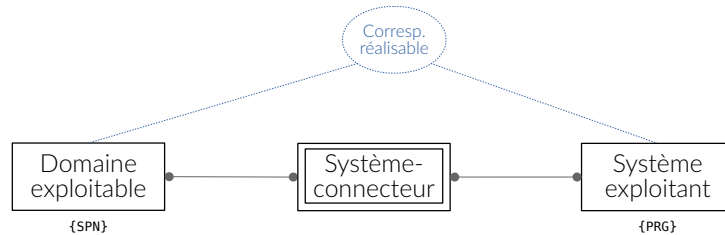


FIGURE C.12 – Problème-archétype : système-connecteur, variante centrée

L'omission du système-connecteur dans l'analyse n'aurait pas permis d'apporter une telle correction. Dans la variante centrée (Figure C.12), c'est le domaine intermédiaire qui doit être conçu. Un cas classique serait une interface homme-machine.

C.3 Catégorisation d'un domaine

La Figure C.13 décrit une méthode rigoureuse pour catégoriser un sous-domaine. On peut se demander si le domaine change au cours du temps. Un CD-ROM par exemple n'a aucune chance d'être modifié dans le temps, à part si on prend en compte l'effet de dégradation. On va alors le catégoriser comme « statique ». Un CD-RW par contre est réinscriptible. Il est donc « dynamique », mais il ne peut se changer lui-même, il est alors « inerte ». Les domaines dynamiques qui réagissent à un stimulus extérieur sont dits « réactifs ». Les autres sont dits « spontanés ».

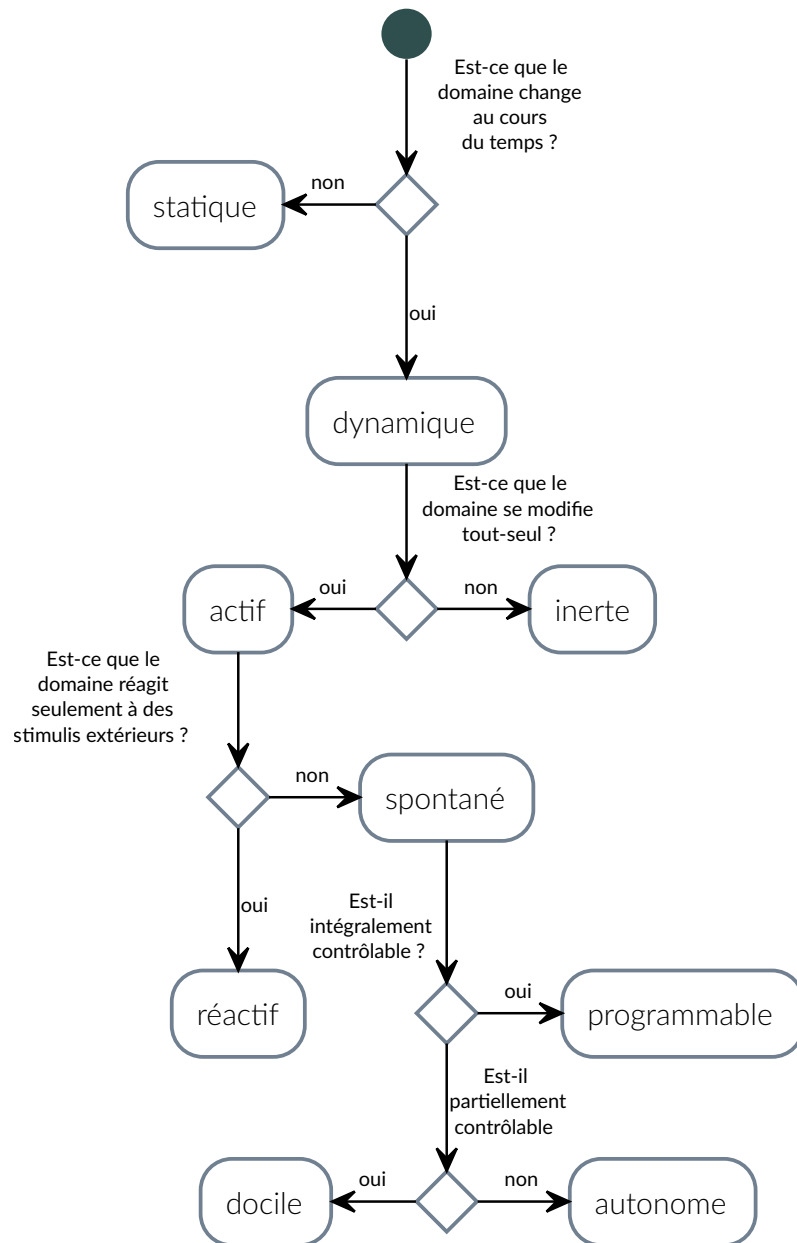


FIGURE C.13 – Catégorisation d'un domaine

Traduit de l'anglais : BRAY, I. K., 2002, page 99 [7]

Suivant leur prédictibilité, ils seront catégorisés comme « programmable » s'ils sont totalement prédictibles, « dociles » s'ils sont partiellement prédictibles, « autonomes » s'ils ne sont absolument pas prédictibles.

En plus des catégories énumérées dans ce diagramme, un domaine peut être « tangible », c'est-à-dire physiquement incarnées (par exemple, un utilisateur, un poste de travail, un routeur,

du *hardware*). Ou bien « intangible », c'est-à-dire existant en dehors de toute enveloppe physique (par exemple, un programme, des règles métier, un **modèle analogique**).

Annexe D

État de l’art

D.1 Cotation du profil des AVQ

TABLEAU D.1 – Cotation du profil des AVQ

Côte	Description
Indépendance	
3 : Indépendance sans difficulté	Capable d’effectuer toutes les opérations de la tâche seul, sans difficulté, dans un délai raisonnable et de façon acceptable. Peut utiliser des aides techniques ou bénéficier d’un environnement adapté.
2 : Indépendance avec difficulté	Capable d’effectuer toutes les composantes de l’opération ou de la tâche seul, mais des difficultés sont observées en ce qui concerne le temps d’exécution ou la façon de réaliser la tâche. Peut utiliser des aides techniques ou bénéficier d’un environnement adapté.
Assistance	
1v : Assistance verbale requise	Capable d’effectuer toutes les composantes de l’opération ou de la tâche avec assistance verbale, dans un délai raisonnable et de façon acceptable ; ceci implique donc un niveau de difficulté suffisamment important pour empêcher l’exécution de la tâche sans intervention de l’examineur. L’aide verbale peut être fournie sous forme de suggestions, encouragements, questions, consignes, indices, avertissements, etc.
1p : Assistance physique requise	Capable d’effectuer toutes les composantes de l’opération ou de la tâche avec assistance physique, dans un délai raisonnable et de façon acceptable ; ceci implique donc un niveau de difficulté suffisamment important pour empêcher l’exécution de la tâche sans intervention de l’examineur. L’aide physique peut être apportée de différentes façons ; soulever la personne lors d’un transfert, installer une aide technique, pousser un fauteuil roulant, actionner les boutons de contrôle du four, donner un appui lors de la marche, etc.

— suite à la page suivante —

Cotation du profil des AVQ – suite

Côte	Description
1vp : Assistance verbale et physique requises	Capable d'effectuer toutes les composante de l'opération ou de la tâche avec assistance verbale et physique. Ceci implique donc un niveau de difficulté suffisamment important pour empêcher l'exécution de la tâche sans intervention de l'examineur.
Dépendance	
0 : Dépendance pour réaliser l'opération ou la tâche	Incapable d'effectuer les composantes de l'opération ou de la tâche dans un délai raisonnable ou de façon acceptable, malgré l'assistance verbale et physique.
Ne sait pas	
6 : La personne interviewée ne peut pas répondre ou n'est pas suffisamment certaine de sa réponse	
Non évalué	
8 : Cause intrinsèque	Opération ou tâche non évaluée pour des raisons intrinsèques à la personne (ex. : refus de coopérer, contre-indication médicale)
9 : Cause extrinsèque	Opération ou tâche non évaluée pour des raisons extrinsèques à la personne. Ces raisons peuvent être : oubli du clinicien, manque de temps, manque d'équipement, manque de ressources humaines, environnement inadapté ou empêchant la mise en situation, etc.

Modifié de : DUTIL, É. et al., 2005, pages 67–69 [17]

D.2 Documentation d'une tâche du profil des AVQ

FIGURE D.1 – Documentation de la tâche « Téléphoner pour une information » du profil des AVQ

D.2.1 Description

Éléments clés de la tâche

- Obtenir une information (horaire d'autobus) par téléphone, c'est-à-dire :
 - trouver le numéro de téléphone du terminus d'autobus ;
 - composer le numéro de téléphone ;
 - demander, obtenir et transmettre l'horaire des départs à l'examineur pour une destination précise.

Matériel requis

- annuaire téléphonique ;
- crayon et papier ;
- formulaire de correction.

Consigne à l'examineur

Demander à la personne de téléphoner pour s'informer de l'horaire des départs d'autobus pour Toronto durant la journée. L'aviser aussi qu'il devra par la suite transmettre l'information à l'examineur.

Note

- Ne mentionner aucune compagnie de transport dans la consigne.
 - La personne peut prendre note du numéro de téléphone et de l'horaire.
 - L'utilisation de l'annuaire téléphonique n'est pas nécessaire ; la personne peut consulter l'assistance annuaire.
 - Ne pas placer l'annuaire, le papier et le crayon devant la personne ; comme pour les autres tâches, elle doit non seulement se les procurer elle-même mais penser à les utiliser.
 - Selon le pays, remplacer Toronto par le nom d'une autre ville.
 - L'opération « formuler un but » ne peut être évaluée avec ce type de scénario.
-

D.2.2 Compte-rendu

Formuler un but

⟨saisie libre⟩

Planifier

⟨saisie libre⟩

Exécuter une tâche

⟨saisie libre⟩

S'assurer de l'atteinte du but initial

⟨saisie libre⟩

Indiquer si les quatre éléments clés mentionnés sont réussis et s'il y a lieu, quels moyens ont été utilisés.

Éléments clés de la tâche

Oui Non

1) trouve le numéro de téléphone du terminus d'auto-bus :

— dans l'annuaire téléphonique ☐ ☐

— par assistance annuaire ☐ ☐

— sur internet ☐ ☐

2) compose le numéro de téléphone : ☐ ☐

3) demande et obtient l'information : ☐ ☐

4) transmet l'information à l'examineur :

— de mémoire ☐ ☐

— après l'avoir écrite ☐ ☐

— après l'avoir imprimée ☐ ☐

Commentaires

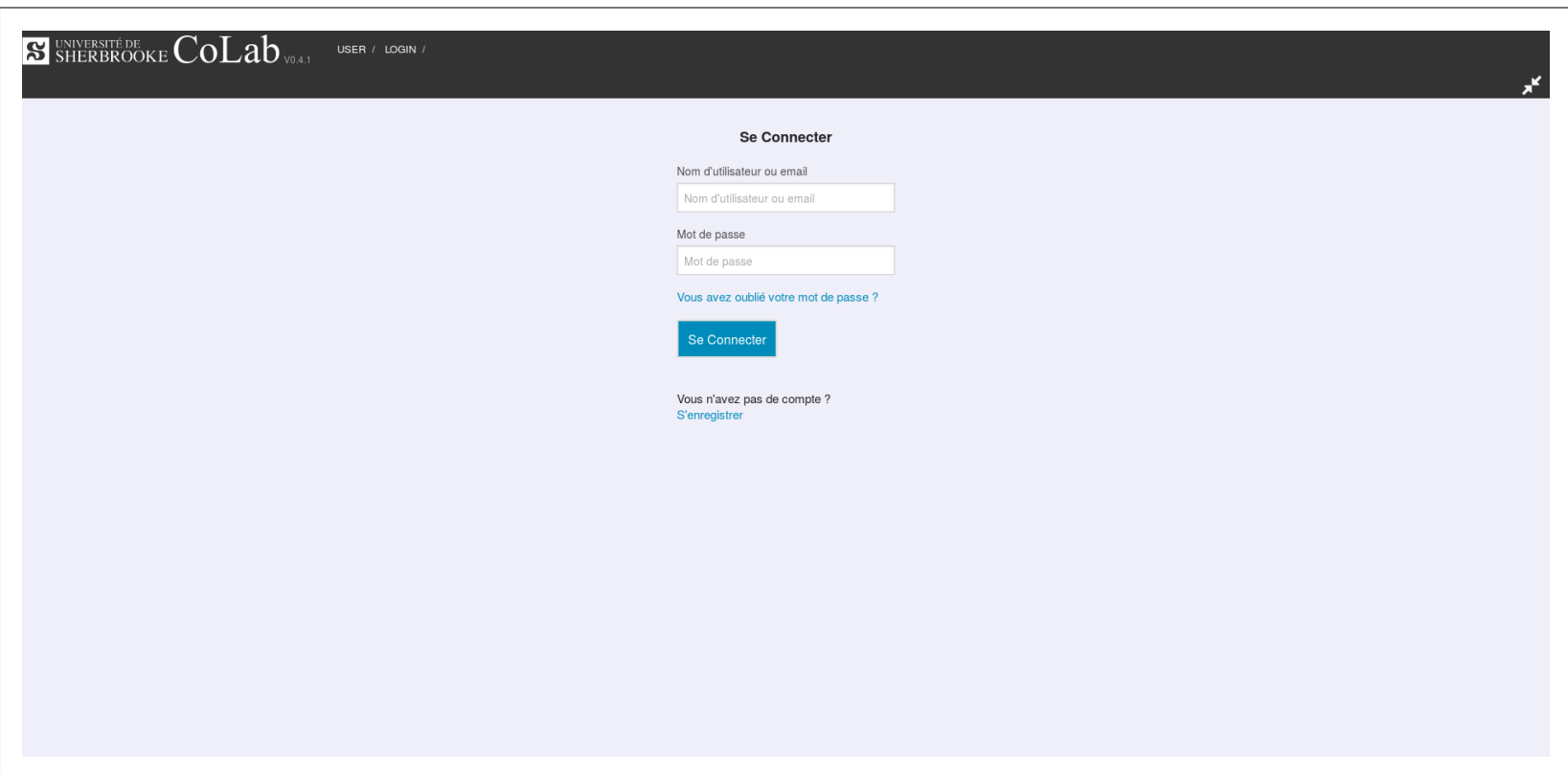
⟨saisie libre⟩

Source : DUTIL, É. et al., 2005, pages 48–51 [17]

Annexe E

Prototypage

E.1 Captures d'écran de CoLAB 0.4



The screenshot shows the login page of the CoLab system. At the top, a dark header bar contains the University of Sherbrooke logo and the text 'UNIVERSITÉ DE SHERBROOKE CoLab V0.4.1' on the left, and 'USER / LOGIN /' on the right. The main content area has a light purple background. In the center, the heading 'Se Connecter' is displayed. Below it are two input fields: 'Nom d'utilisateur ou email' and 'Mot de passe'. A link 'Vous avez oublié votre mot de passe ?' is positioned below the password field. A blue 'Se Connecter' button is centered below the links. At the bottom, the text 'Vous n'avez pas de compte ?' is followed by a blue 'S'enregistrer' link.

UNIVERSITÉ DE SHERBROOKE CoLab V0.4.1 USER / LOGIN /

Se Connecter

Nom d'utilisateur ou email

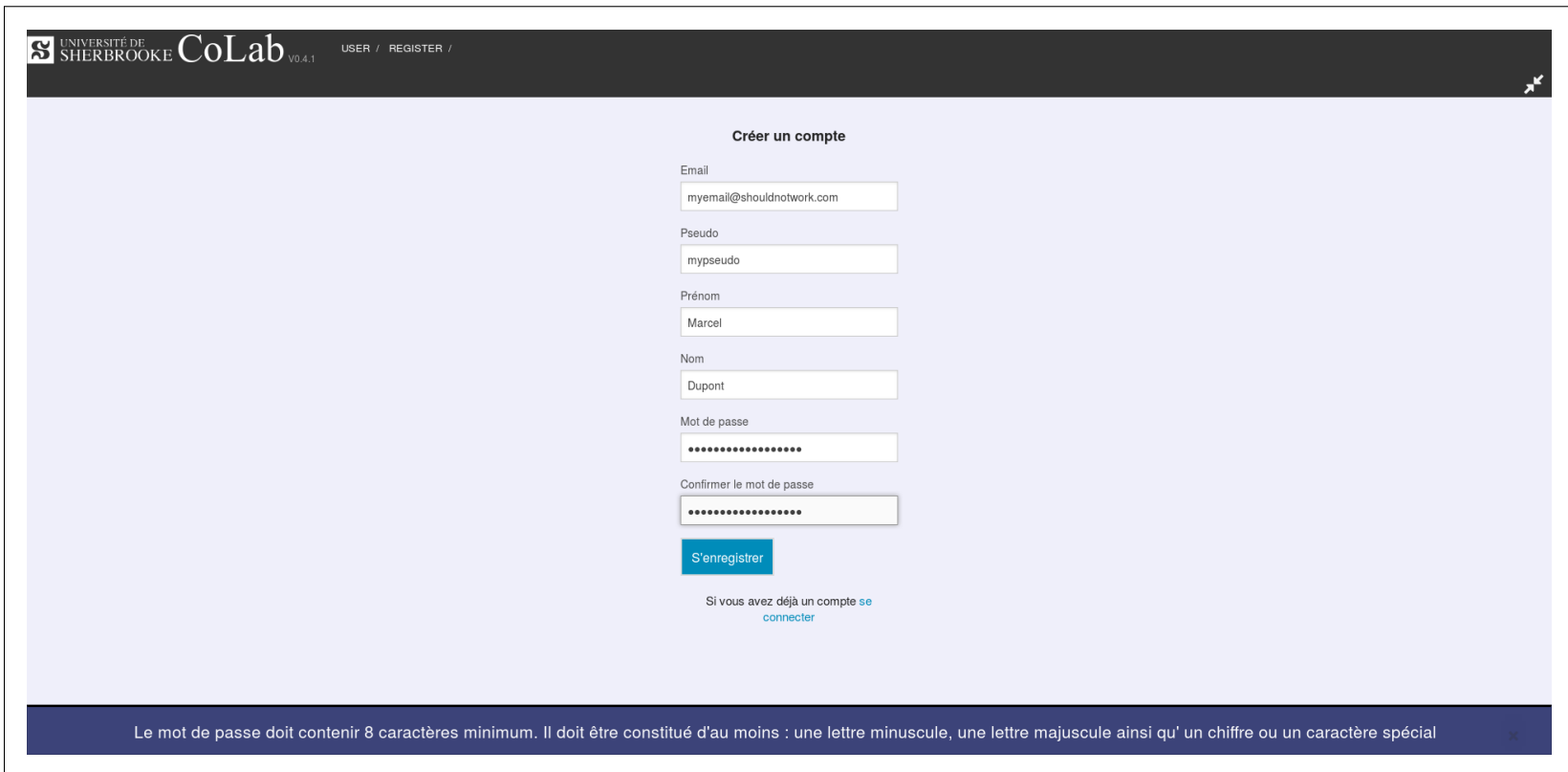
Mot de passe

[Vous avez oublié votre mot de passe ?](#)

[Se Connecter](#)

[Vous n'avez pas de compte ?](#)
[S'enregistrer](#)

FIGURE E.1 – Capture d'écran de la page d'authentification /user/login



The screenshot shows a web browser window with a dark header bar. On the left, the logo for 'UNIVERSITÉ DE SHERBROOKE CoLab' is displayed, with 'V0.4.1' and 'USER / REGISTER /' to its right. On the far right of the header is a small icon of two arrows pointing away from each other. The main content area has a light blue background and is titled 'Créer un compte'. It contains a vertical stack of input fields: 'Email' (with 'myemail@shouldnotwork.com'), 'Pseudo' (with 'mypseudo'), 'Prénom' (with 'Marcel'), 'Nom' (with 'Dupont'), 'Mot de passe' (with masked dots), and 'Confirmer le mot de passe' (also with masked dots). Below these fields is a blue button labeled 'S'enregistrer'. Underneath the button is a link that says 'Si vous avez déjà un compte [se connecter](#)'. At the bottom of the page, a dark blue footer bar contains the text: 'Le mot de passe doit contenir 8 caractères minimum. Il doit être constitué d'au moins : une lettre minuscule, une lettre majuscule ainsi qu' un chiffre ou un caractère spécial'.

UNIVERSITÉ DE SHERBROOKE CoLab V0.4.1 USER / REGISTER /

Créer un compte

Email
myemail@shouldnotwork.com

Pseudo
mypseudo

Prénom
Marcel

Nom
Dupont

Mot de passe
.....

Confirmer le mot de passe
.....

S'enregistrer

Si vous avez déjà un compte [se connecter](#)

Le mot de passe doit contenir 8 caractères minimum. Il doit être constitué d'au moins : une lettre minuscule, une lettre majuscule ainsi qu' un chiffre ou un caractère spécial

FIGURE E.2 – Capture d'écran de la page de création de compte /user/register, visuel par défaut

UNIVERSITÉ DE SHERBROOKE CoLab V0.4.1 USER / REGISTER /

Créer un compte

Mot de passe: Doit contenir au moins 1 lettre minuscule, 1 lettre majuscule, 1 chiffre ou un 1 caractère spécial
Confirmer le mot de passe: Doit contenir au moins 1 lettre minuscule, 1 lettre majuscule, 1 chiffre ou un 1 caractère spécial

Email
myemail@shouldnotwork.com

Pseudo
mypseudo

Prénom
Marcel

Nom
Dupont

Mot de passe
.....

Confirmer le mot de passe
.....

Accès prohibé. Contacter l'administrateur du site.

FIGURE E.3 – Capture d'écran de la page de création de compte /user/register, mot de passe invalide et adresse courriel absente de la liste blanche

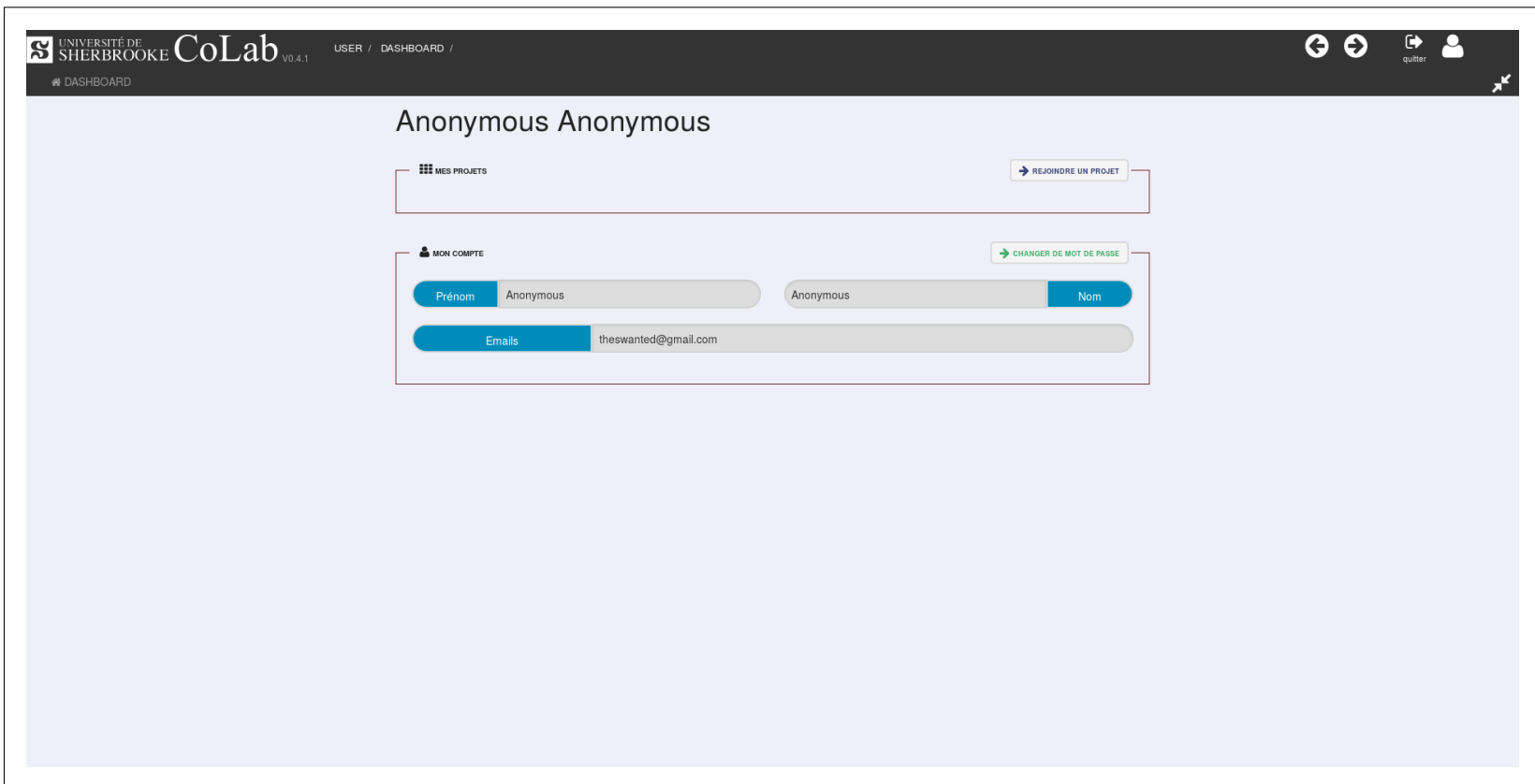


FIGURE E.4 – Capture d'écran de la page de tableau de bord /user/dashboard

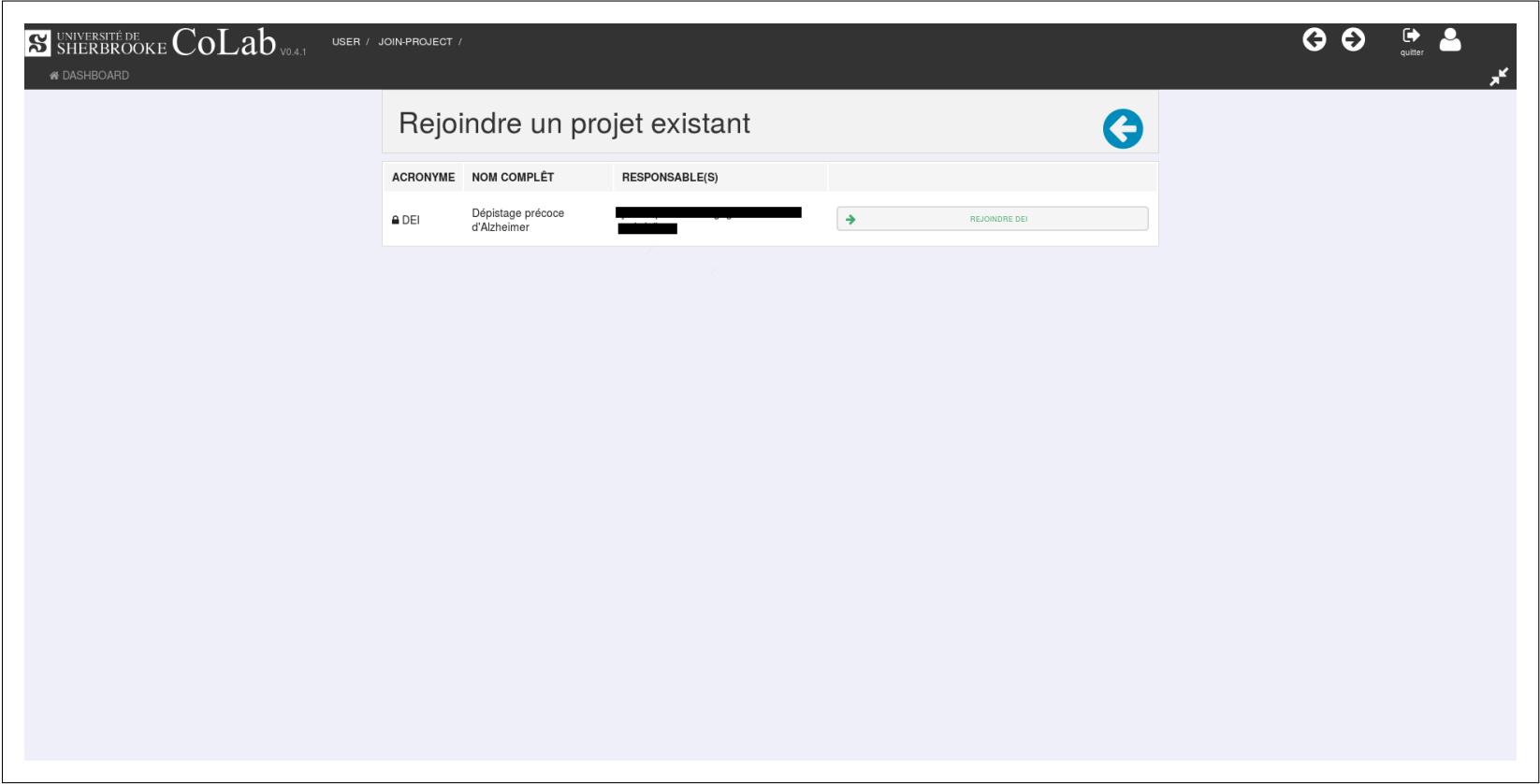


FIGURE E.5 – Capture d’écran de la page d’intégration aux projets /user/join-project

UNIVERSITÉ DE SHERBROOKE CoLab V0.4.1 PROJECTS / DEI /

DASHBOARD O DEI

DEI Dépistage précoce d'Alzheimer

EXPÉRIMENTATIONS

25 expérimentation(s) trouvée(s)

COLLABORATION	NON D'EXP ANONYME	DATE	DURÉE
[ANON] 2234		2014-12-03	38m08s
[ANON] 0887		2014-12-10	35m44s
[ANON] 7653		2014-12-10	49m45s
[ANON] 1562		2015-01-16	33m04s
[ANON] 5428		2015-01-16	45m15s
[ANON] 2098		2015-01-30	26m51s
[ANON] 0953		2015-01-30	36m42s
[ANON] 1087		2015-02-06	38m13s
[ANON] 2307		2015-03-06	49m32s
[ANON] 7382		2015-03-06	30m11s
[ANON] 3223		2015-05-06	48m46s
[ANON] 9032		2015-05-06	73m55s

EXPÉRIMENTATIONS
INFO/ACTIONS
MEMBRES

FIGURE E.6 – Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un associé

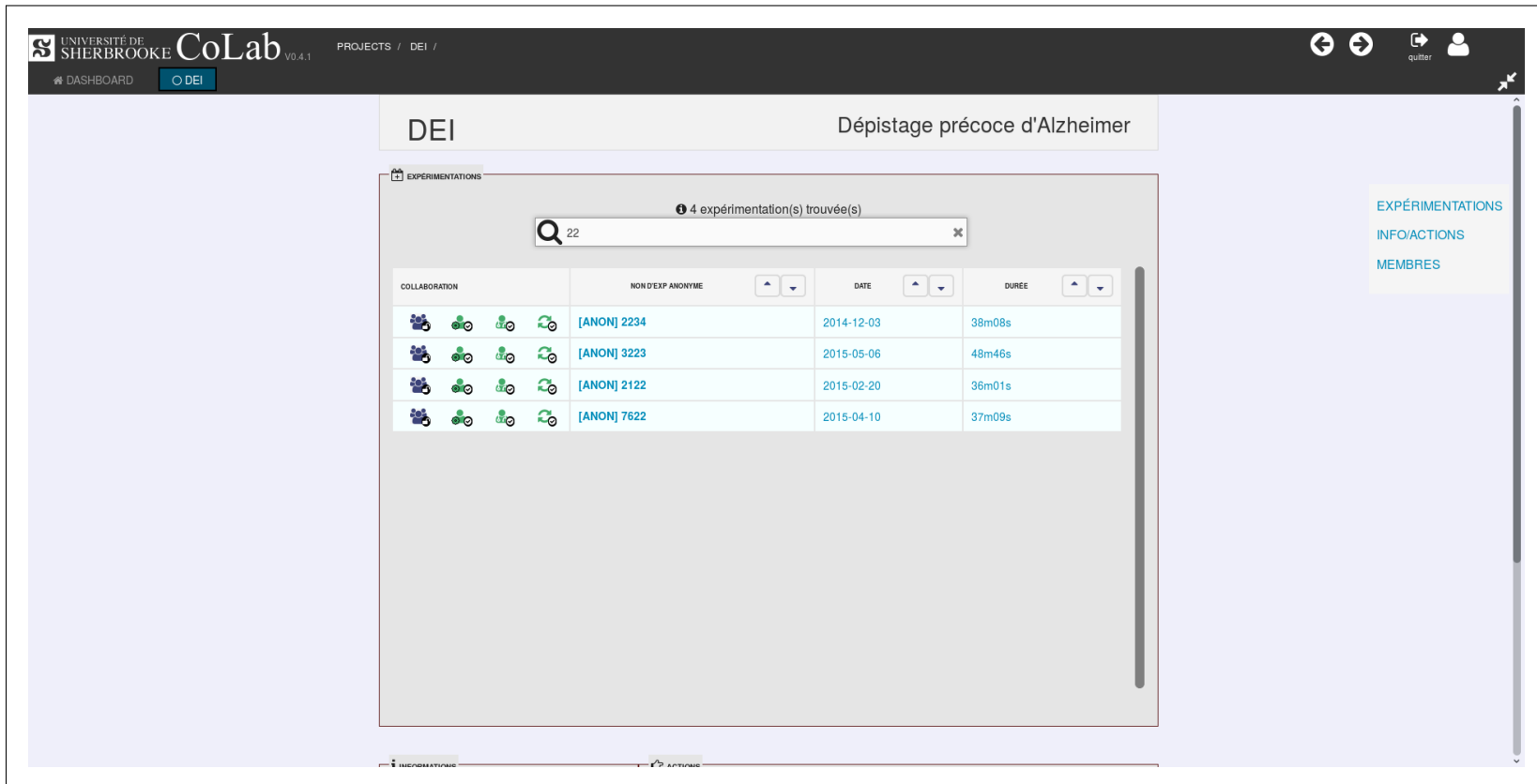


FIGURE E.7 – Capture d'écran de la page d'accueil du projet **DEI** /projects/DEI, visuel sur l'explorateur d'expérimentations par un associé, avec saisie d'un filtre textuel

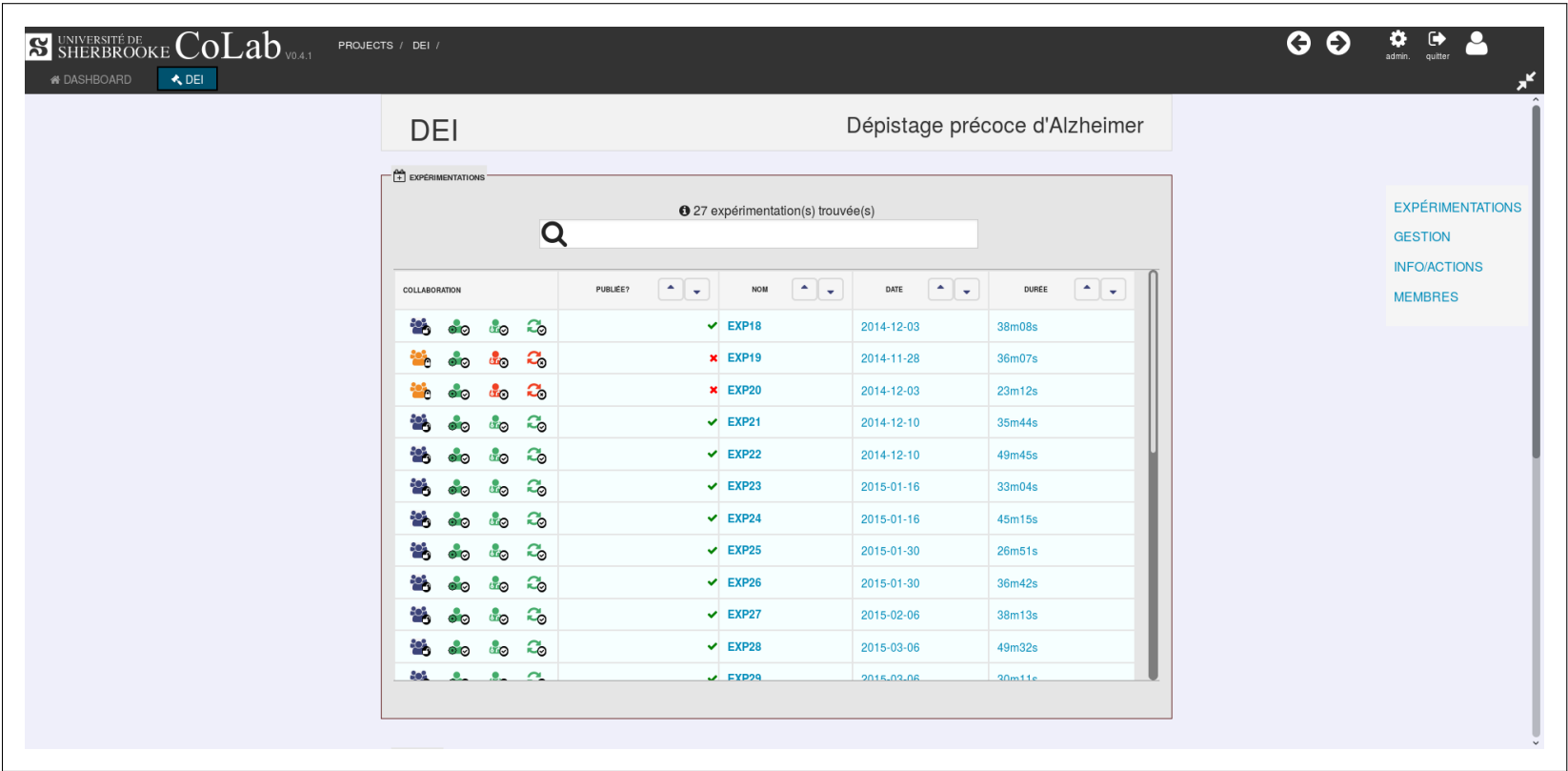


FIGURE E.8 – Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur l'explorateur d'expérimentations par un responsable

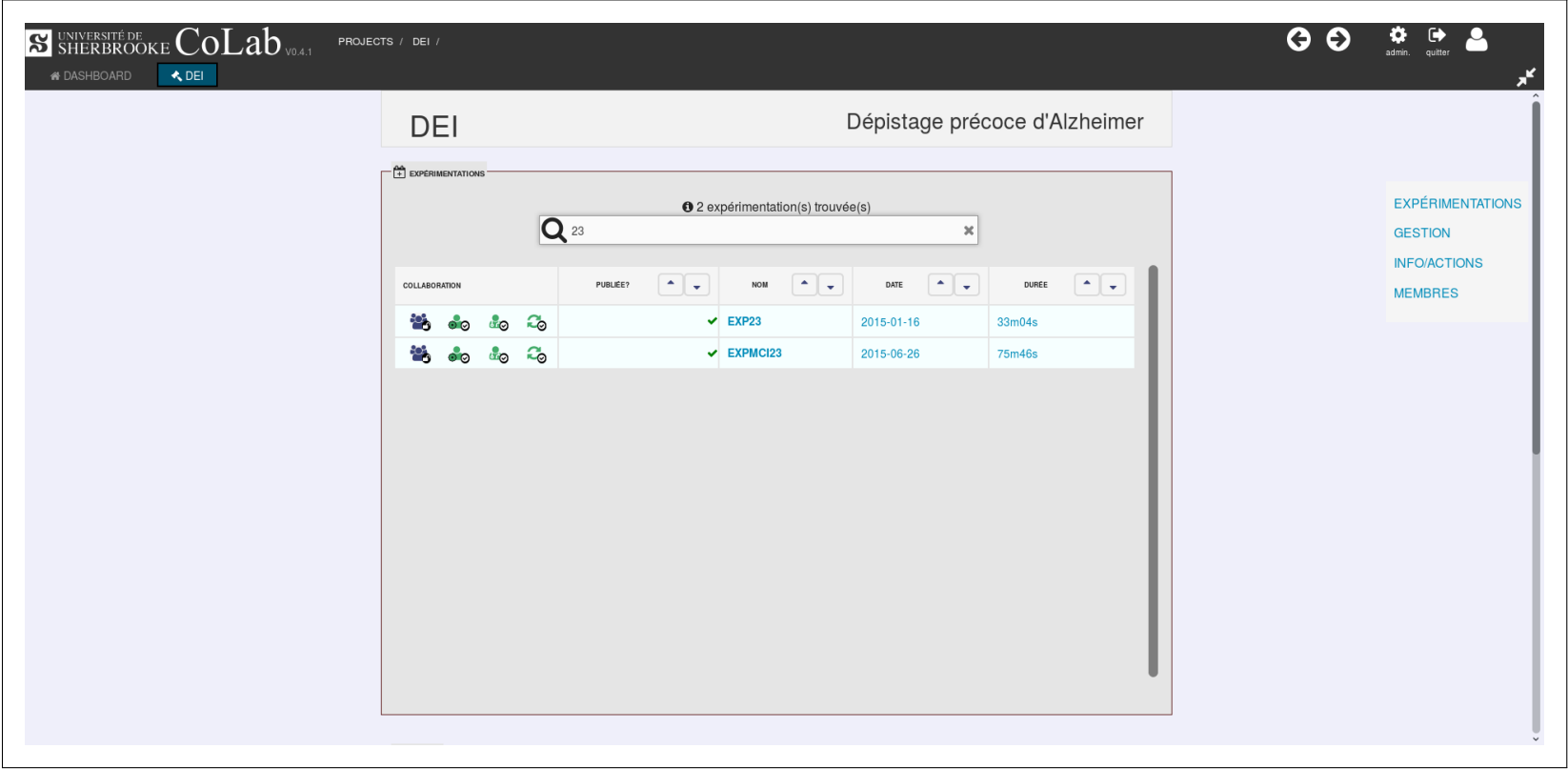


FIGURE E.9 – Capture d’écran de la page d’accueil du projet **DEI** /projects/DEI, visuel sur l’explorateur d’expérimentations par un associé, avec saisie d’un filtre textuel

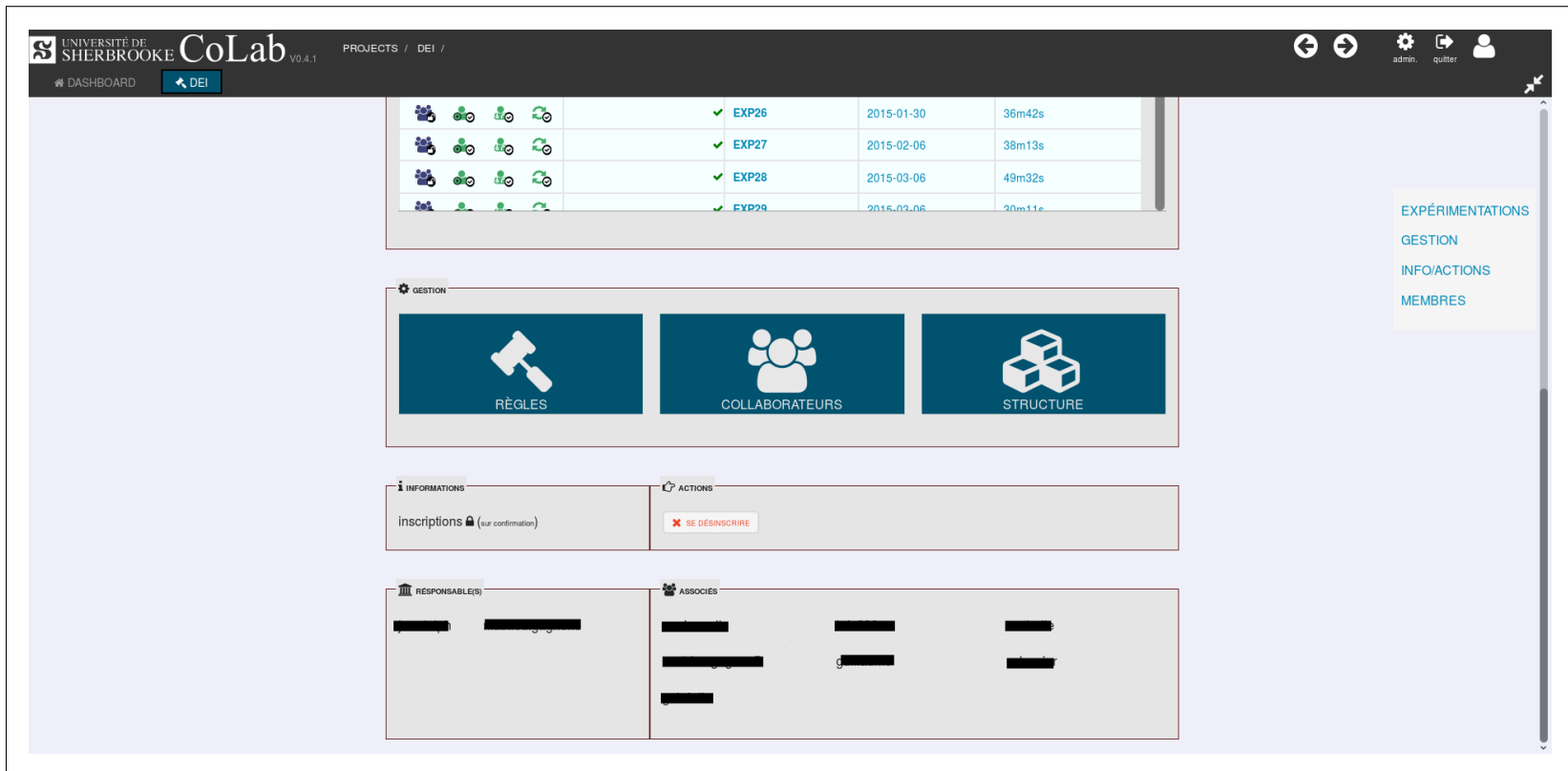


FIGURE E.10 – Capture d'écran de la page d'accueil du projet DEI /projects/DEI, visuel sur le panneau de configuration par un responsable

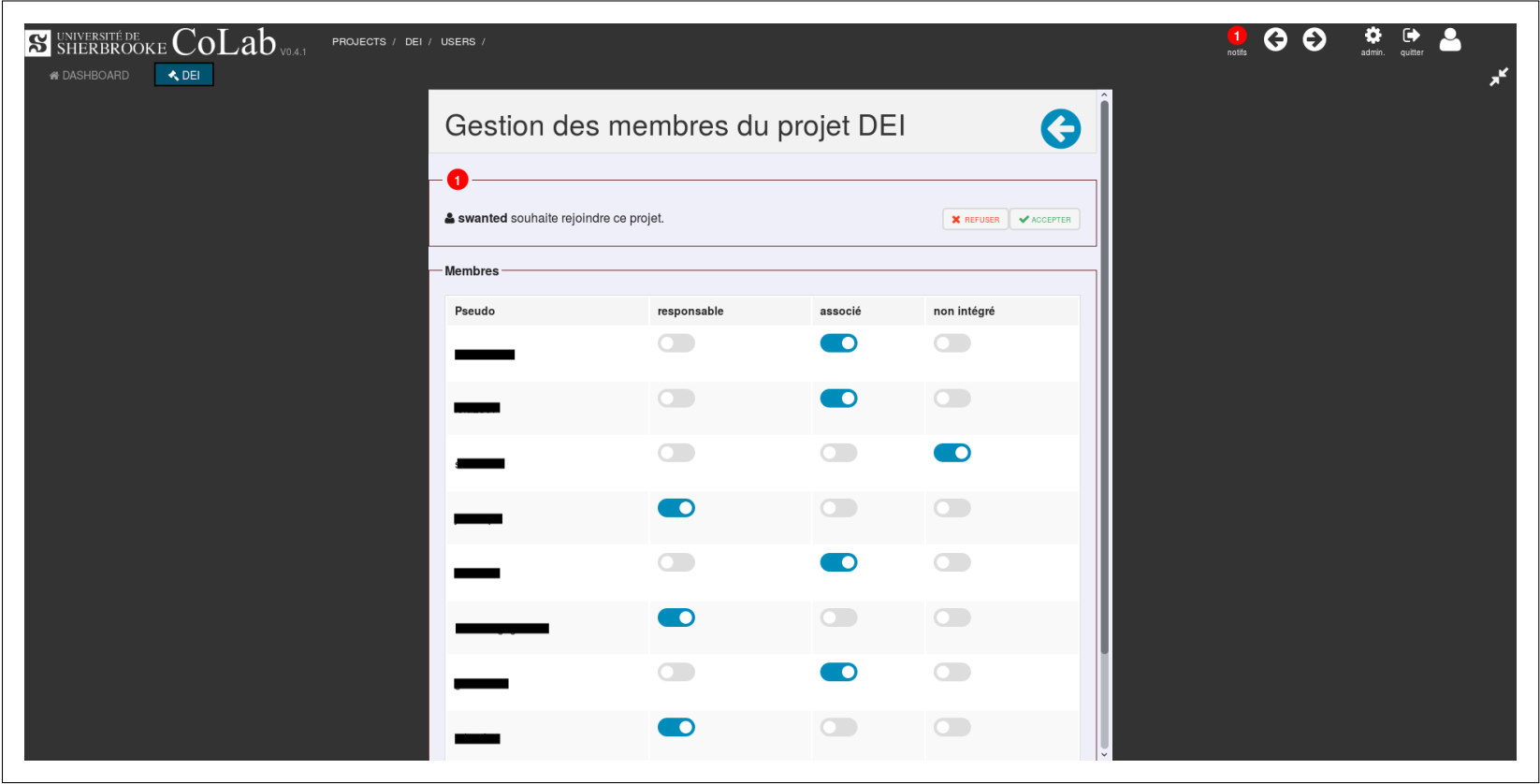


FIGURE E.11 – Capture d’écran de la page de gestions des membres d’un projet /projects/DEI/users, une demande d’intégration au projet est en suspend

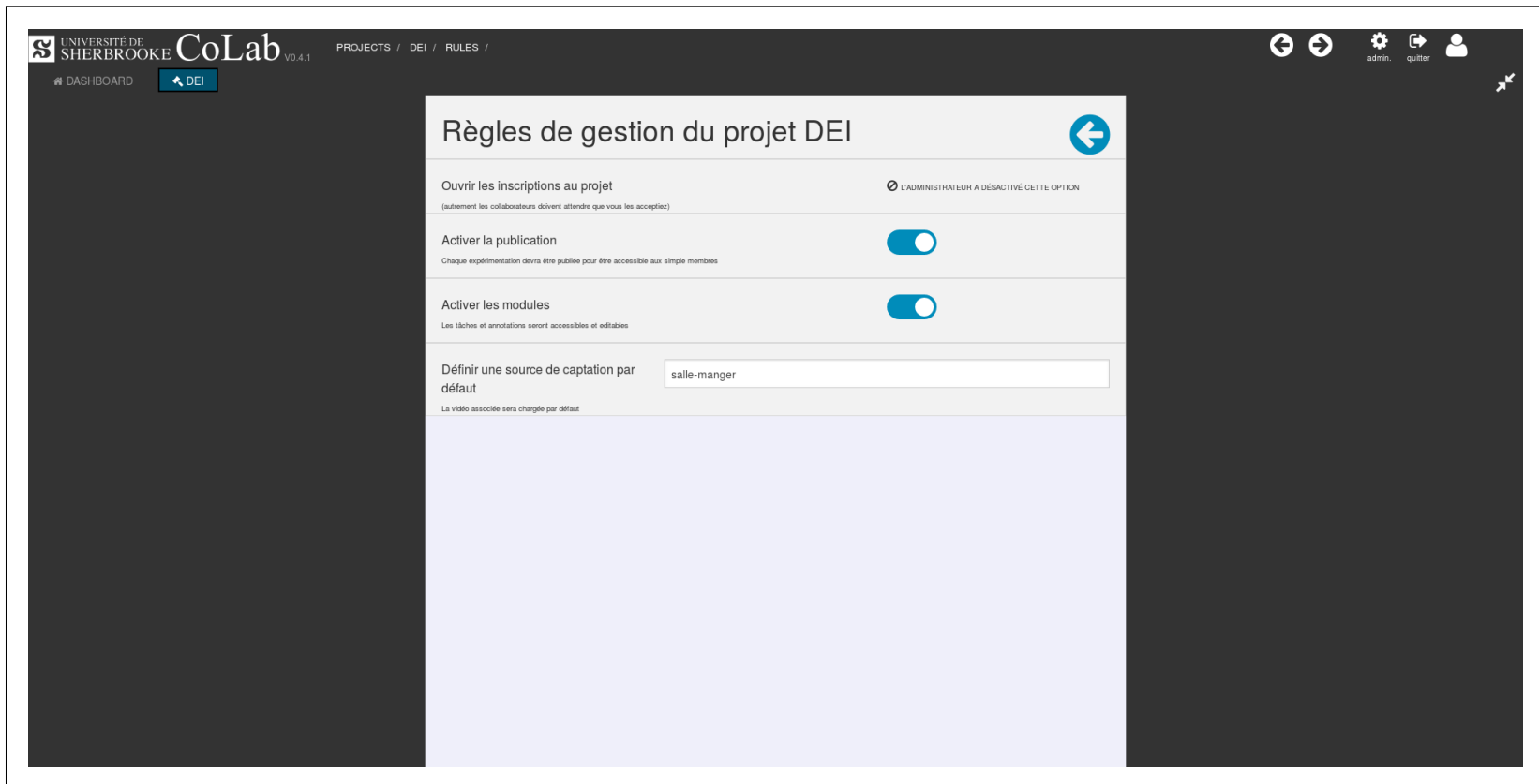


FIGURE E.12 – Capture d'écran de la page de configuration des règles d'un projet /projects/DEI/rules

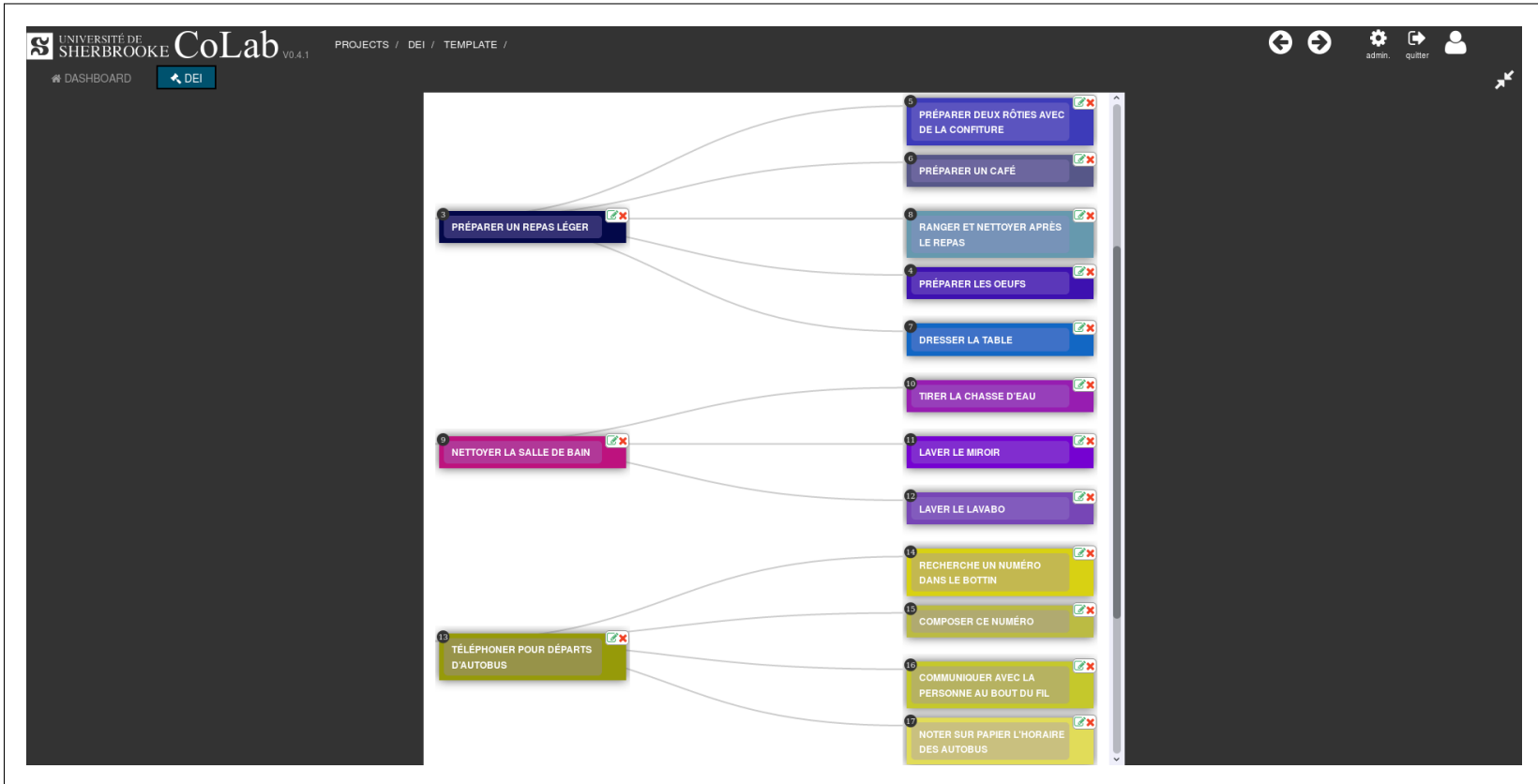


FIGURE E.13 – Capture d'écran de la page de définition d'un *template* de projet /projects/DEI/template, visuel sur l'arborescence de tâches

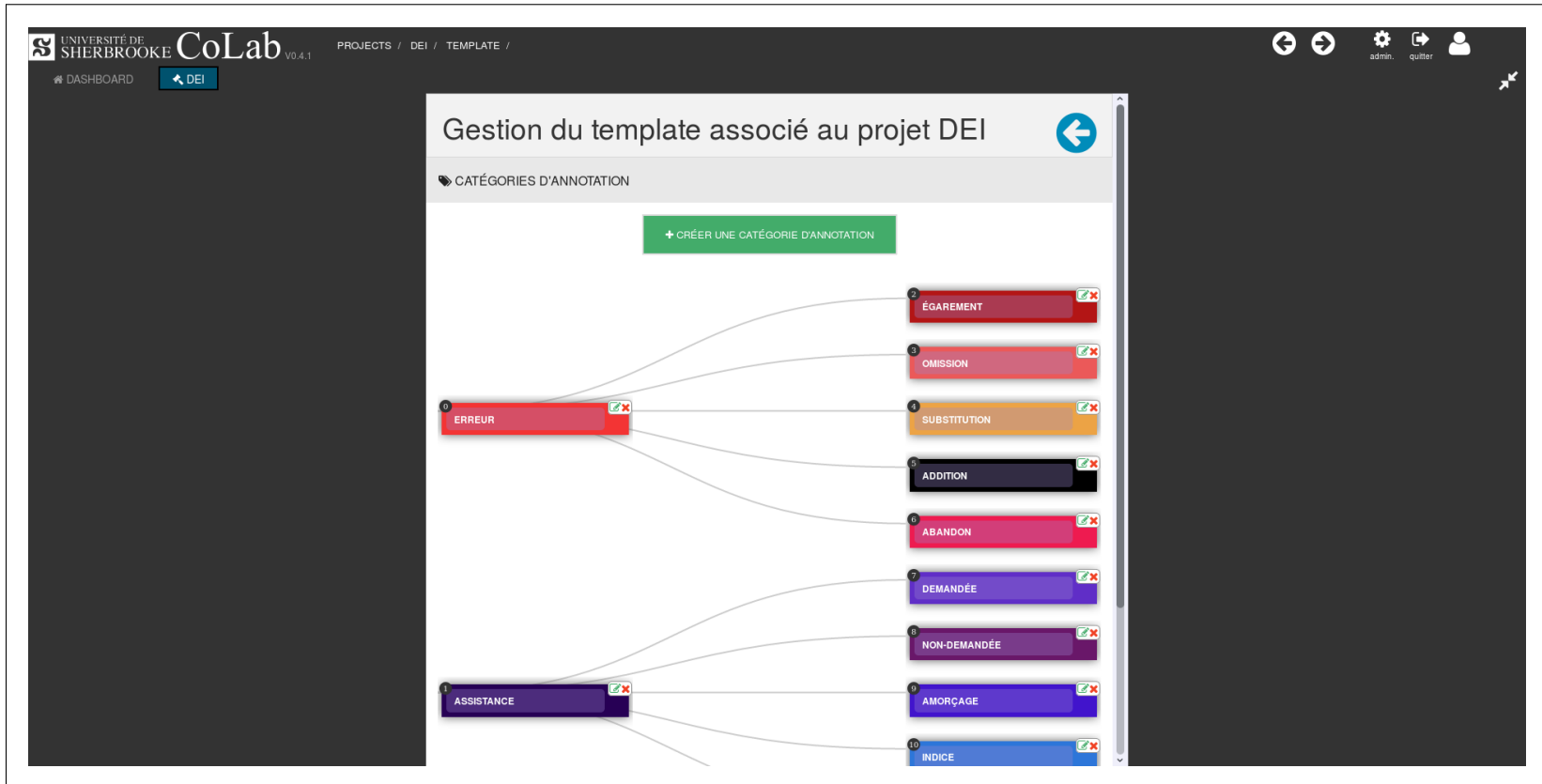


FIGURE E.14 – Capture d'écran de la page de définition d'un *template* de projet /projects/DEI/template, visuel sur l'arborescence d'annotations

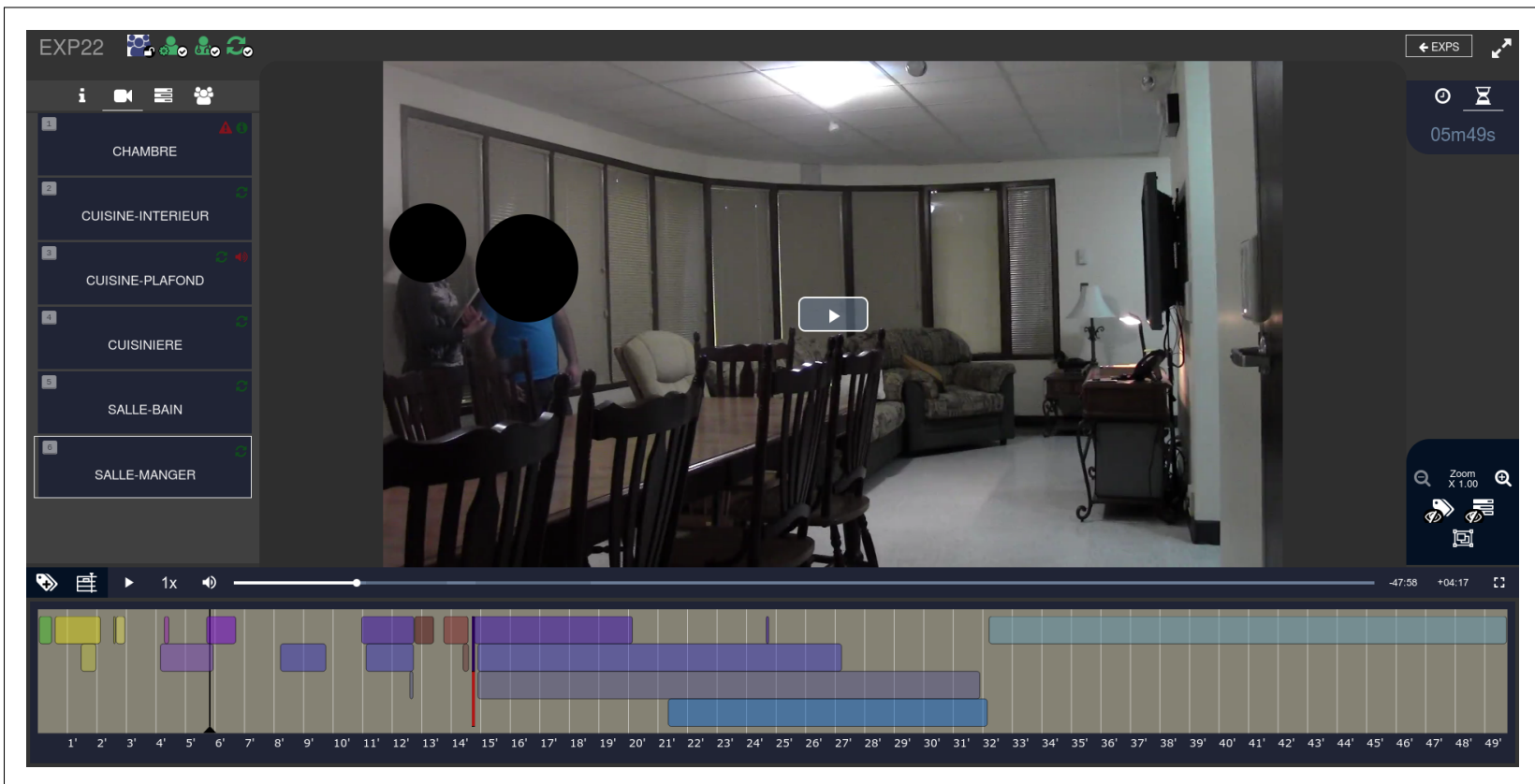


FIGURE E.15 – Capture d'écran de la page d'expérimentation `/projects/DEI/exps/56a6982e4c7cbf05907982f7`, visuel par défaut

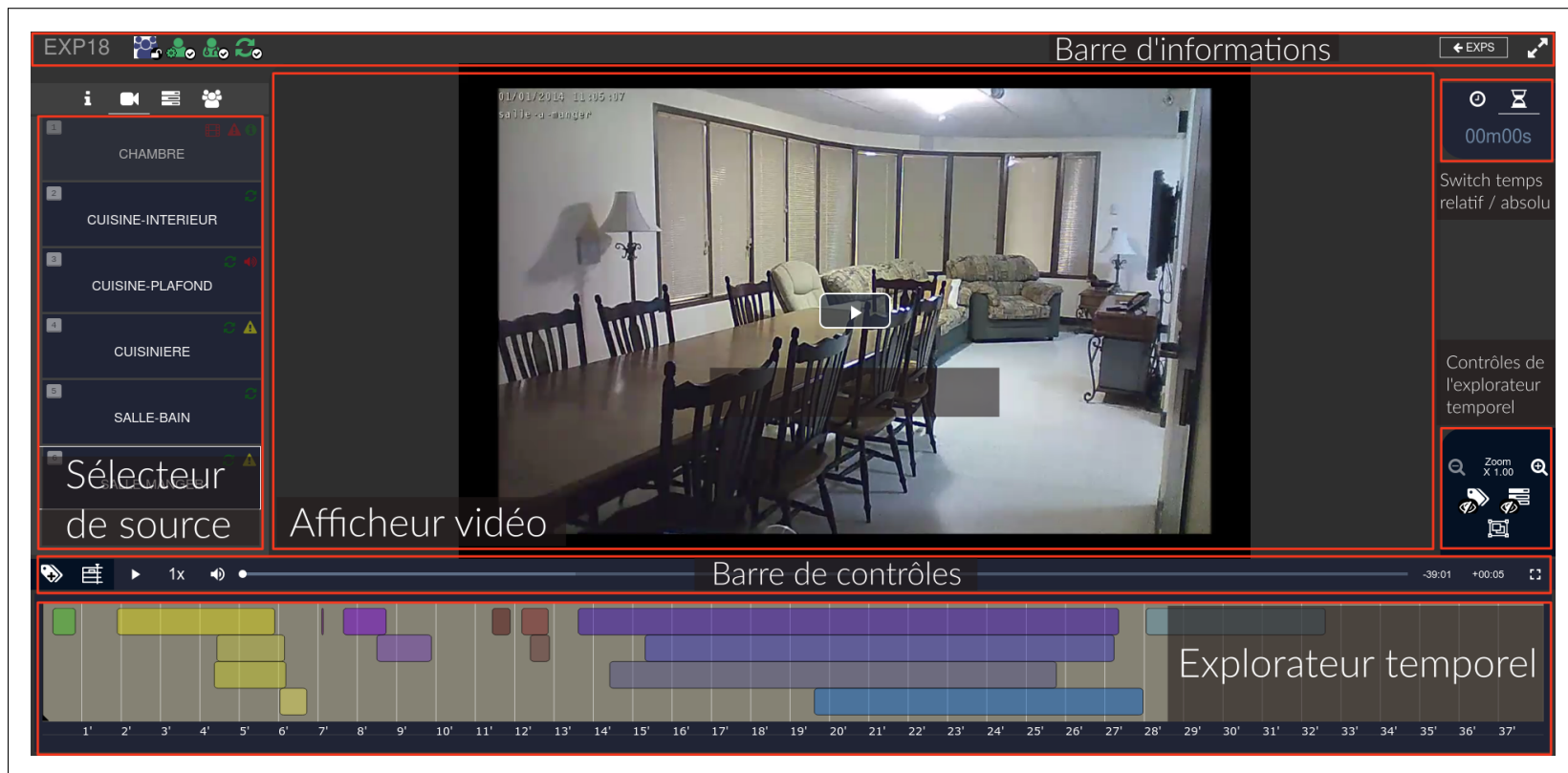


FIGURE E.16 – Capture d'écran de la page d'expérimentation `/projects/DEI/exps/56a6982e4c7cbf05907982f7`, visuel par défaut

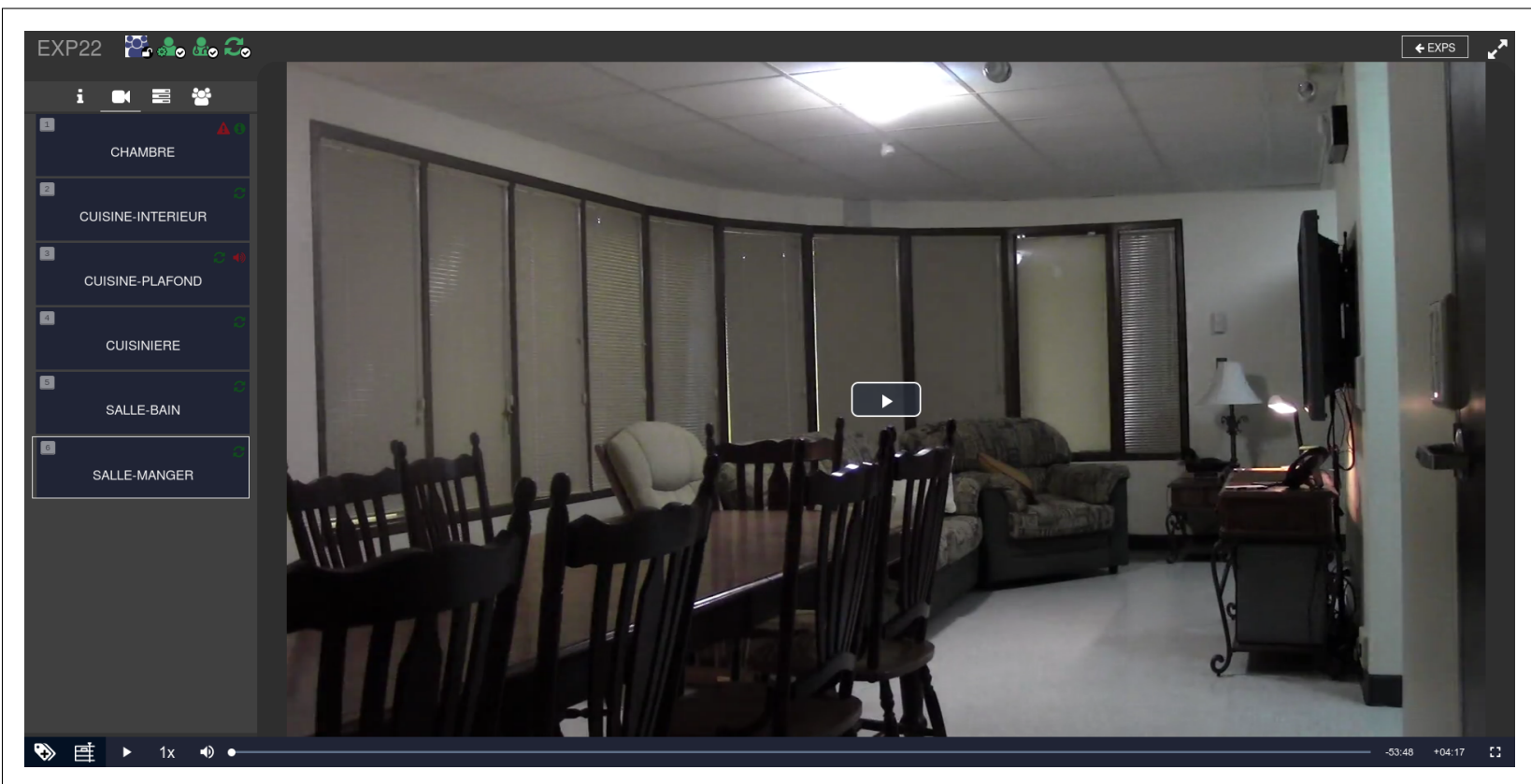


FIGURE E.17 – Capture d'écran de la page d'expérimentation, visuel étendu

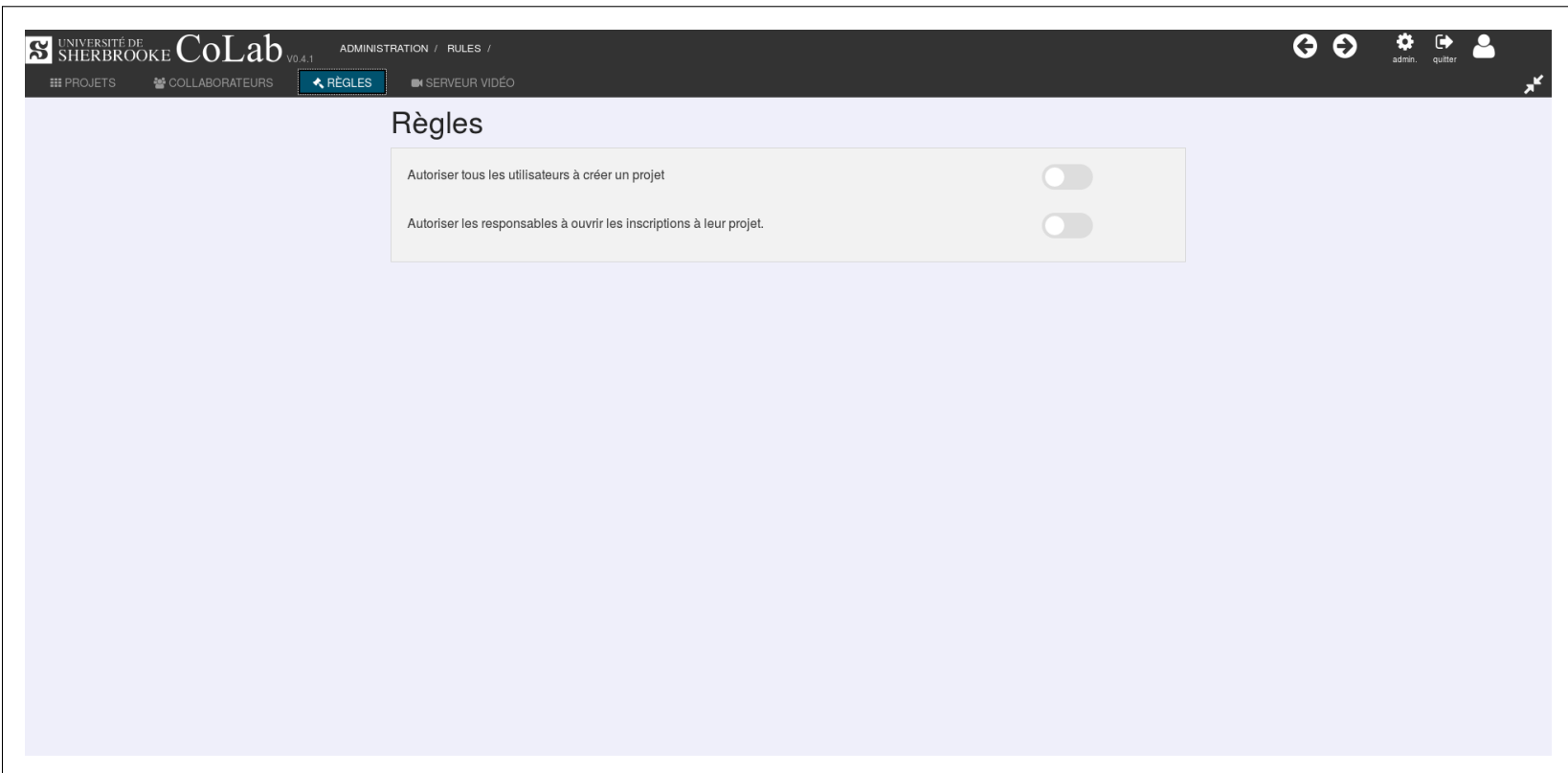


FIGURE E.18 – Capture d'écran de la page d'administration des règles /admin/rules

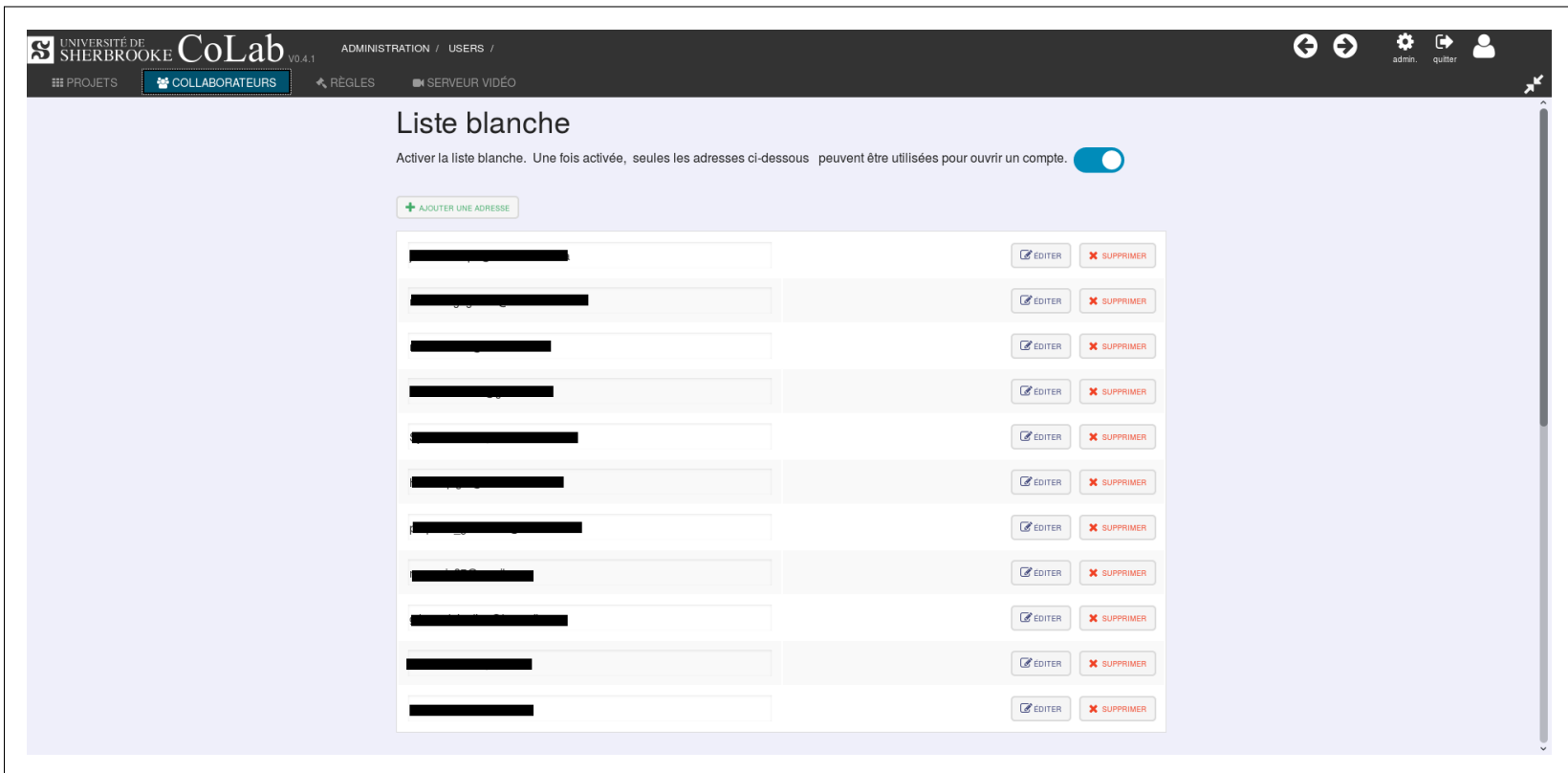


FIGURE E.19 – Capture d'écran de la page d'administration des utilisateurs /admin/users, visuel sur la liste blanche

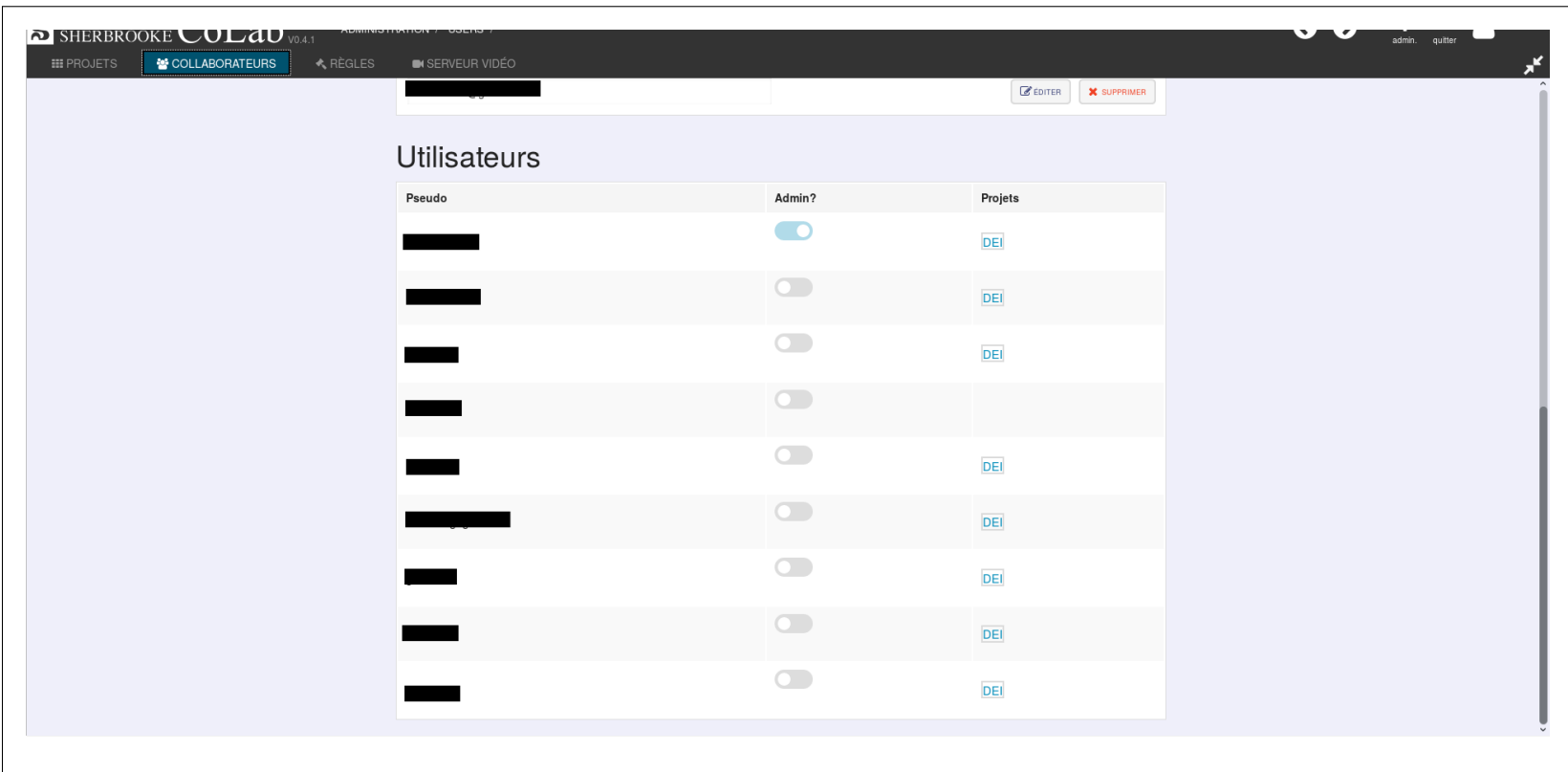


FIGURE E.20 – Capture d'écran de la page d'administration des utilisateurs /admin/users, visuel sur la liste d'utilisateurs

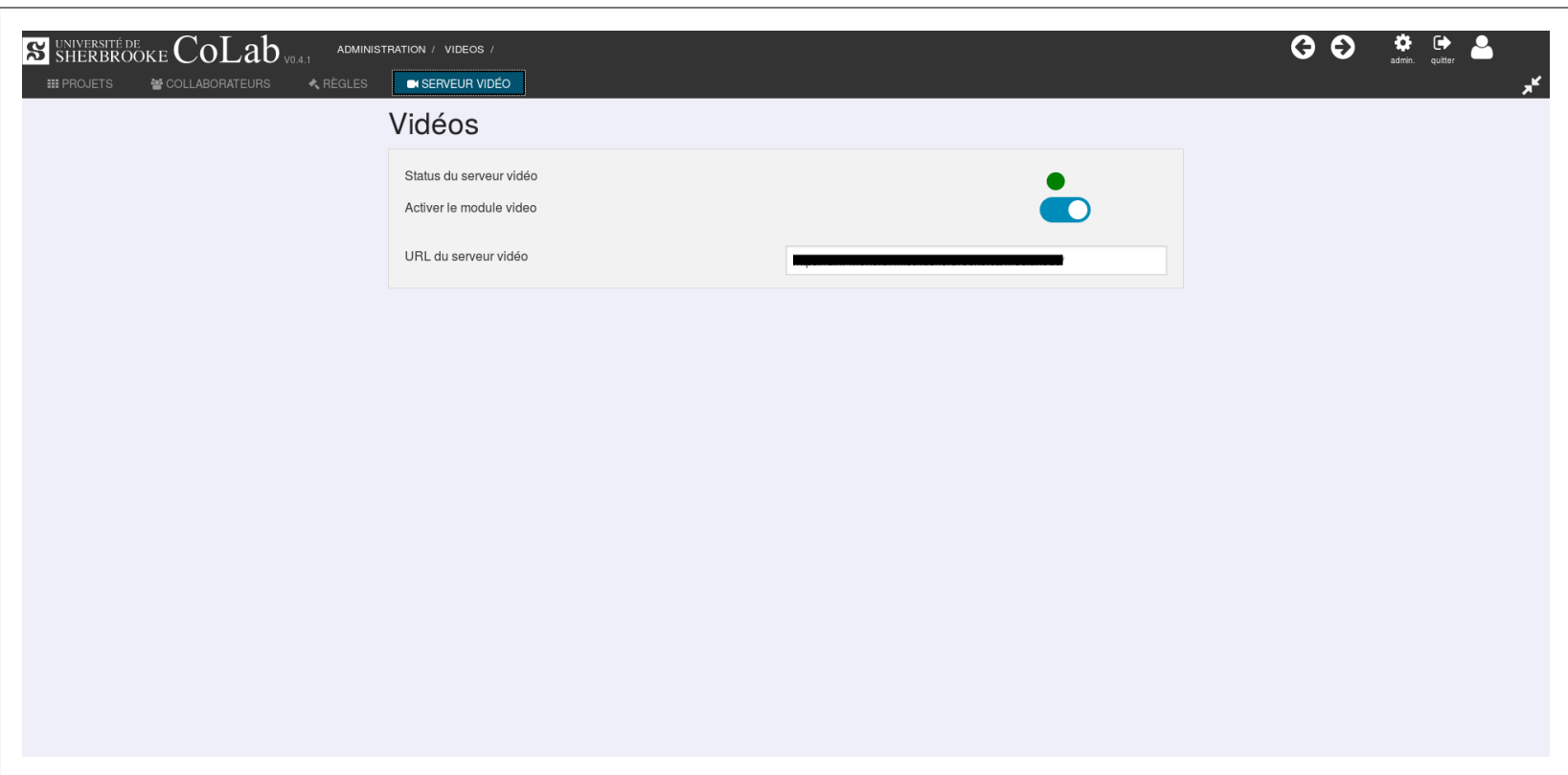


FIGURE E.21 – Capture d'écran de la page d'administration du serveur vidéo /admin/videos

E.2 Captures d'écran de COLAB 0.5

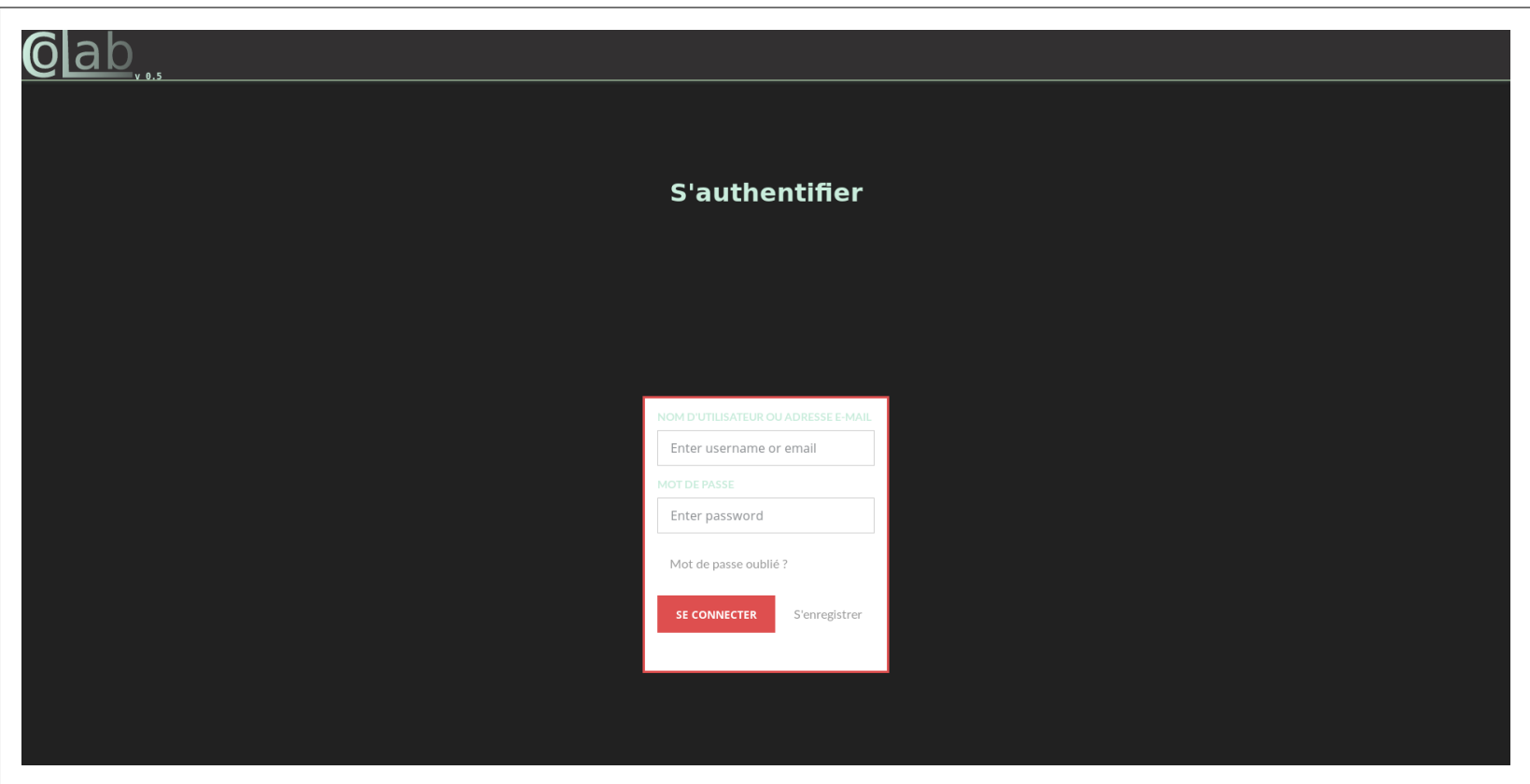


FIGURE E.22 – Capture d'écran de la page d'authentification / 1

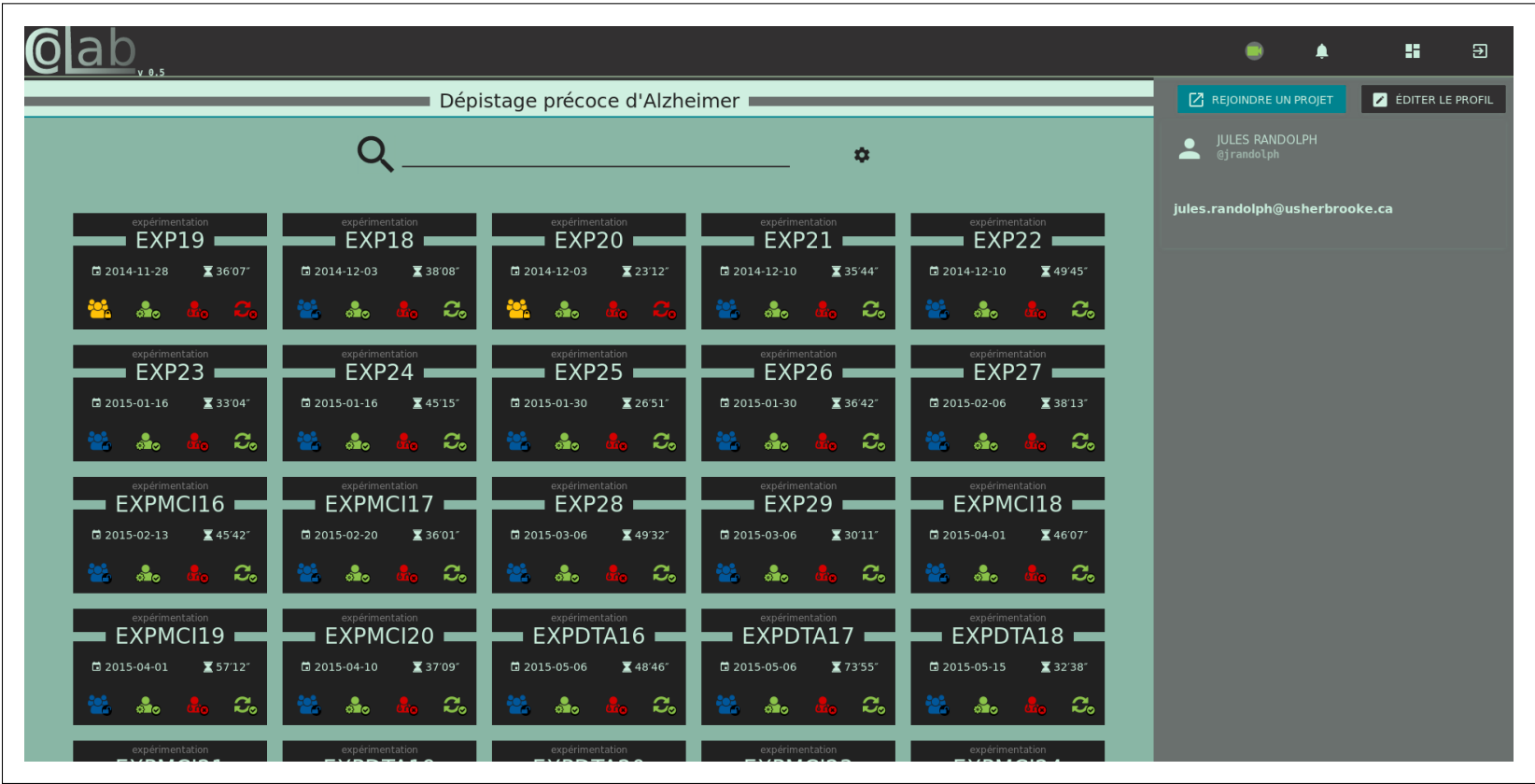


FIGURE E.23 – Capture d'écran de la page de tableau de bord /d



FIGURE E.24 – Capture d'écran de la page relative à l'expérimentation EXP26 du projet DEI /e/DEI/EXP20, sans explorateur temporel

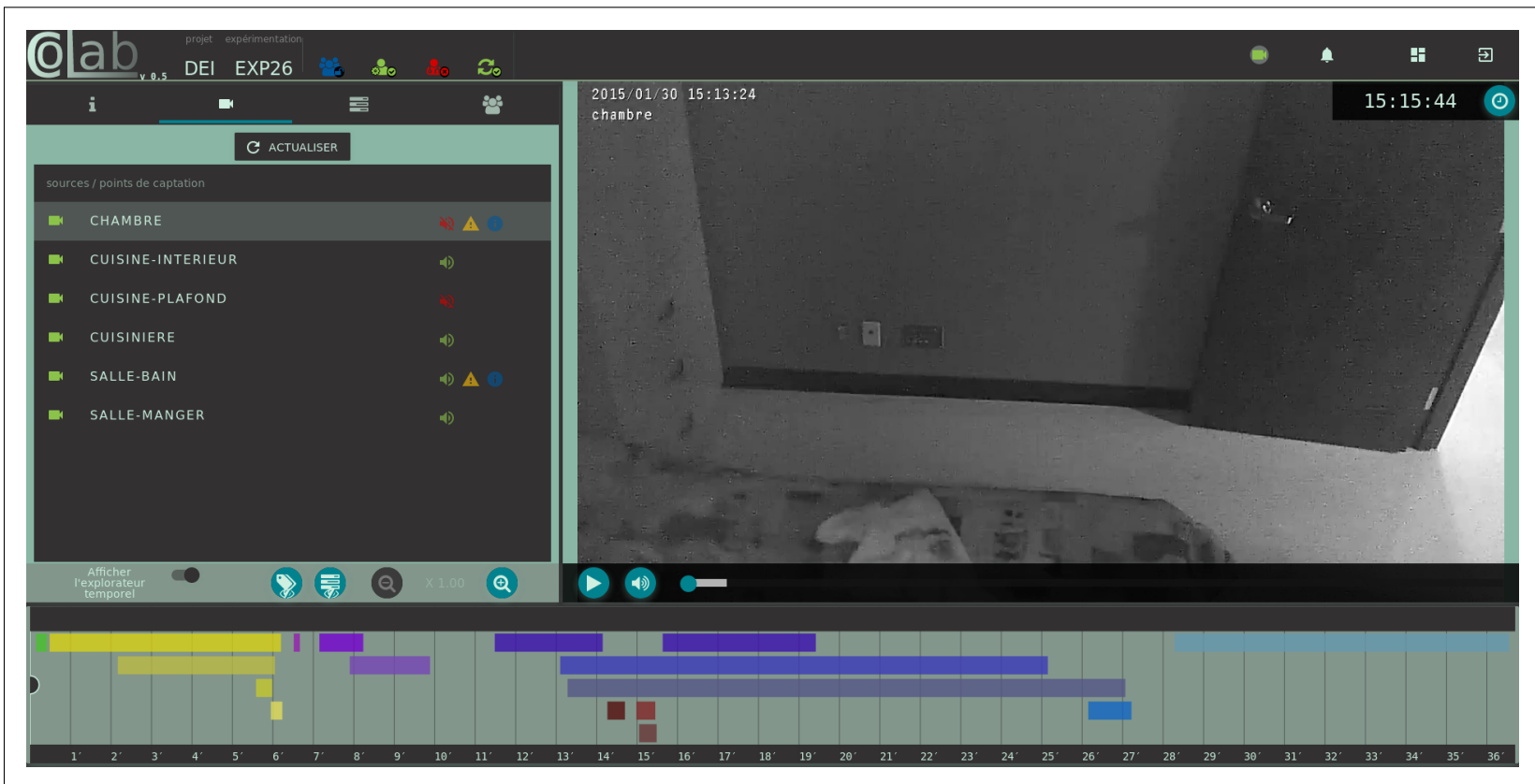


FIGURE E.25 – Capture d'écran de la page relative à l'expérimentation EXP26 du projet DEI /e/DEI/EXP20, avec explorateur temporel

E.3 Serrurier

E.3.1 Implémentation du décorateur « cadenas »

Cet extrait de code est issu de la librairie Serrurier, accessible à l'adresse donnée en référence [62].

```

1  import { ensures } from './ensures';
2  import { propagateException } from './api/security';
3  import { buildAssertion } from './api/Cadenas';
4  import { Match } from 'meteor/check';
5  import { ActionsStore } from 'meteor/svein:serrurier-core';
6  /**
7   * @param name
8   * @param {...*} params - params to give to the named cadenas
9   * @example
10  * @cadenas( 'persisted' )
11  * methodThatMustBeRunWhenPersisted(){
12  *     // method content
13  * }
14  *
15  */
16  export default function cadenas(name, ...params ) {
17      let thatMethodName = '';
18      let assertion = buildAssertion( () => thatMethodName, name, params );
19      return function cadenasDecoratorApplier( target, methodName ) {
20          const oldAction = target[methodName];
21          ensures( 'The annotation `cadenas` must target a function, but found a '+typeof
22          ↪ oldAction, oldAction, Function );
23          thatMethodName = methodName;
24          target[methodName] = function cadenasRunner() {
25              /** @type error_descriptor */
26              let errorDescriptor = assertion.perform( this, arguments );
27              // if at least one assertion returns an error descriptor, store it and stop
28              ↪ proceeding assertions.
29              if( Match.test( errorDescriptor, Object ) ) {
30                  try {
31                      propagateException.call( this, errorDescriptor );
32                  } catch(error){
33                      error.context = Object.assign( {}, errorDescriptor );
34                      throw error;
35                  }
36              }
37              else {
38                  try {
39                      return oldAction.apply( this, arguments );
40                  } catch(error){
41                      error.context = Object.assign( {}, error.error, error.reason );
42                      throw error;
43                  }
44              }
45          };
46          ActionsStore.registerOrUpdate( target[methodName], methodName );
47          target[methodName]._cadenasMethod=methodName;
48          return target[methodName];
49      }
50  }

```


E.3.2 Implémentation du cadenas « `loggedUserInRole` »

Cet extrait de code est issu de la librairie `serrurier-cadenas-roles`, accessible à l'adresse suivante : <https://github.com/sveinburne/serrurier-cadenas-roles>.

```

1  import { DefaultCadenas } from 'meteor/svein:serrurier';
2  import { Roles } from 'meteor/alanning:roles';
3  import { Match } from 'meteor/check';
4  import {
5    isUserInRole_s,
6    isLoggedUserInRole_s,
7  } from './roles-utils';
8
9  //noinspection JSValidateTypes
10 /**
11  * @desc A scope that resolves with this.getScope()
12  * @type {role_scope}
13  */
14  const AUTO = {};
15
16  /**
17  * @desc A scope that resolves to the global scope
18  * @type {role_scope}
19  */
20  //noinspection JSUnresolvedVariable
21  const GLOBAL = Roles.GLOBAL_PARTITION;
22
23
24  /**
25  * Assert logged user is in [roles]|role, partition
26  */
27  const loggedUserInRole = new DefaultCadenas({
28    name: 'loggedUserInRole',
29    doesAssertionFails: function ( role_s, scope = GLOBAL ) {
30      if( scope === AUTO ) scope = this.getScope();
31      const userNotInRole = !isLoggedUserInRole_s( role_s, scope );
32      return userNotInRole && `user.not.in.role:${role_s}:${scope}`
33    },
34    matchPatterns: [
35      // role_s
36      Match.OneOf( String, [String] ),
37      // scope
38      Match.OneOf( String, Match.Where((val) => val === GLOBAL || val === AUTO ) )
39    ],
40    dependingCadenas: { userIsLoggedIn: [] }
41  });
42
43  const parts = {
44    GLOBAL,
45    AUTO
46  };
47
48  export {
49    parts
50  }

```

E.3.3 Exemple de définition d'une classe avec contrôle d'accès

```

1  import '/imports/init-behaviors'
2  import '/imports/init-conf-cadenas'
3  import { Mongo } from 'meteor/mongo'
4  import {
5    Serrurier,
6    cadenas,
7    server
8  } from 'meteor/svein:serrurier'
9  import valid from '../validation'
10 import { roles } from '../security'
11 import { Plugins } from './Plugins'
12 import includes from 'lodash/includes'
13 import { Meteor } from 'meteor/meteor'
14 import { Roles } from 'meteor/alanning:roles'
15 import Config from './Config'
16 import { parts } from 'meteor/svein:serrurier-cadenas-roles'
17
18 const projects = new Mongo.Collection('projects')
19
20 /**
21  * Core entity gathering a set of collaborators and a set of experiments.
22  * @class
23  * @implements {ScopeClass}
24  */
25 const Project = Serrurier.createClass({
26   name: 'Project',
27   collection: projects,
28   behaviors: {
29     scopeRoot: {
30       // Adds #getScope method
31     }
32   },
33   methods: {
34     /**
35      * @return {boolean} - Whether or not logged user is willing to join this project.
36      */
37     @cadenas('userIsLoggedIn')
38     isLoggedInUserPending() {
39       return includes(this.pending, Meteor.userId())
40     },
41     /**
42      * @return {Object[]} - A list of users which are waiting for a privilege update.
43      */
44     @cadenas('loggedUserInRole', roles.project$acceptMember, parts.AUTO)
45     getPendingUsers() {
46       return Meteor.users.find({
47         _id: {
48           $in: this.pending
49         }
50       }).fetch()
51     },
52     /**
53      * @desc Update user privileges in project.
54      * @param {!string} userId
55      * @param {!string} role - 'MEMBER' or 'MANAGER' or 'NONE'
56      * @param {Function} asyncCallback - function (error, result) {...}
57      */
58     @server()
59     @cadenas('loggedUserInRole', roles.project$updateUserRole, parts.AUTO)
60     updateUserRoleInProject(userId, role, asyncCallback = null) {

```

```

61     if (role === 'MEMBER') {
62       this.setUserRoles(userId, roles.project$MEMBER)
63     } else if (role === 'MANAGER') {
64       this.setUserRoles(userId, roles.project$MANAGER)
65     } else if (role === 'NONE') {
66       this.setUserRoles(userId, [])
67     }
68   },
69
70   /**
71    * @desc Add logged user to the pending list.
72    * @memberof Project#
73    * @instance
74    */
75   @server()
76   @cadenas('userIsLoggedIn')
77   @cadenas('loggedUserNotInRole', roles.project$MEMBER, parts.AUTO)
78   registerToPendingInProject() {
79     Project.update(this._id, {
80       $push: {
81         pending: Meteor.userId()
82       }
83     })
84   },
85   /**
86    * @desc Cancel a request to join a project
87    * @param {?string=} userId - The unique identifier of the user.
88    * Default value (if null or undefined) : Meteor.userId()
89    * @param {Function=} asyncCallback - function (error, result) {...}
90    */
91   @server()
92   @cadenas('userIsLoggedIn')
93   cancelPendingInProject(userId, asyncCallback = null) {
94     Project.update(this._id, {
95       $pull: {
96         pending: Meteor.userId()
97       }
98     })
99   }
100 },
101 fields: {
102   /**
103    * @type {String}
104    * @desc String of maximum 10 chars, minimum 3 chars
105    */
106   acronym: {
107     type: String,
108     validators: [valid.minLength(3, 10)]
109   },
110   /**
111    * @type {String}
112    * @desc String of maximum 300 chars, minimum 10 chars
113    */
114   fullName: {
115     type: String,
116     validators: [valid.minLength(10, 300)]
117   },
118   /**
119    * @type {Plugins}
120    */
121   plugins: {
122     type: Plugins,

```

```

123     default: () => new Plugins()
124   },
125   /**
126    * @type {boolean}
127    */
128   isOpen: {
129     type: Boolean,
130     default: false
131   },
132   /**
133    * @type {boolean}
134    */
135   publicationPolicy: {
136     type: Boolean,
137     default: true
138   },
139   /**
140    * Whether or not plugins (annotations and tasks) should be enabled.
141    * Can be resolved to enableModules
142    * @type {boolean}
143    * @default false
144    */
145   enablePlugins: {
146     type: Boolean,
147     default: false
148   },
149   /**
150    * @type {string}
151    * @default null
152    */
153   defaultCaptionSource: {
154     type: String,
155     default: null,
156     optional: true
157   },
158   /**
159    * A list of pending user ids that wish to join this project
160    * @type {string[]}
161    * @default []
162    */
163   pending: {
164     type: [String],
165     default: []
166   }
167 },
168 indexes: {
169   uniqAcronym: {
170     fields: {
171       acronym: 1
172     },
173     options: {
174       unique: true
175     }
176   }
177 }
178 })
179
180 export default Project

```

E.4 medianode

E.4.1 Gestionnaire d'authentification pour CoLAB

```

1  import _ from "lodash"
2  import { MongoClient as MClient } from "mongodb"
3
4  export default function(uri, options, Promise) {
5
6    return {
7      assertCredentials(credentials) {
8        const areCredentialsWellFormatted = _.isString(credentials.user) &&
9          ⇨ _.isString(credentials.hash)
10         if(areCredentialsWellFormatted) {
11           this.logger.trace(`well-formatted credentials`)
12         } else {
13           this.logger.warn(`ill-formatted credentials`)
14         }
15         this.logger.trace(`asserting credentials : ${areCredentialsWellFormatted}`)
16         return areCredentialsWellFormatted
17       },
18
19       isUserInProject(userId, projectAcronym) {
20         const promise = new Promise(function(accept, reject) {
21           return MClient.connect(uri).then(db => {
22             let users = db.collection('users')
23             let projects = db.collection("projects")
24             return projects.findOne({acronym: projectAcronym, _id:1}).then(prj =>
25               ⇨ users.findOne({
26                 _id: userId,
27                 roles: {
28                   _id: "project.member",
29                   partition: `projects.${prj.acronym}`
30                 }
31               })))
32           .then(function(user) {
33             db.close()
34             return accept(user !== null)
35           })
36         })
37         return promise
38       },
39
40       authClientASync(credentials) {
41         const promise = new Promise(accept => {
42           return MClient.connect(uri).then(db=> {
43             let selector = {_id:credentials.user}
44             let users = db.collection('users')
45             return users.findOne(selector).then(user=> {
46               db.close()
47               if (user) {
48                 const hasCredits = (_.get(user, "services.password.bcrypt")) ===
49                   ⇨ credentials.hash
50                 if (hasCredits) {
51                   this.logger.info(`user ${credentials.user} access granted`)
52                 } else {
53                   this.logger.warn(`user ${credentials.user} access denied`)
54                 }
55                 return accept(hasCredits)
56               } else {
57

```

```
54         this.logger.warn(`user ${credentials.user} is not registered!`)
55         return accept(false)
56     }
57     })
58   })
59 })
60 return promise
61 }
62 }
63 }
```

Annexe F

Questionnaire d'utilisation de CoLAB

F.1 Questionnaire brut

Le questionnaire peut être consulté dans le [Tableau F.1](#). Pour des raisons de place, aucune des images jointes aux questions n'a été intégré à cette table. De même, seules les descriptions obligatoires à sa compréhension ont été retranscrites. Une question précédée du symbole « ★ » revêt un caractère obligatoire. Quant aux formats de réponse possible, il en existe quatre. Les saisies de texte et de nombre sont identifiées respectivement avec les motifs *⟨saisie libre⟩* et *⟨nombre⟩*. Les échelles de Likert sont retranscrites assez explicitement de la manière qui suit :

valeur 1 ○○○○○○○ valeur 2

Enfin, les réponses des questions à choix sont retranscrites avec un carré ☐ lorsque plusieurs choix sont possibles, avec un rond ☐ lorsqu'un seul choix est possible.

TABLEAU F.1 – Formulaire d'utilisation de CoLAB

Code	Question	Format de réponse
Préambule		
Renseignements personnels		
P.Q01	★ Autorisez-vous qu'on reprenne contact avec vous pour obtenir des précisions sur votre perception de l'outil ?	<input type="radio"/> oui <input type="radio"/> non
P.Q02	Quel est votre âge ?	<i>⟨nombre⟩</i>
Mise en contexte		
P.Q03	★ Comment évalueriez-vous votre confort avec les outils numériques ?	Pas du tout confortable ○○○○○○○ Très confortable
P.Q04	★ Mon / mes domaine(s) d'expertise sont ...	<input type="checkbox"/> Ergothérapie <input type="checkbox"/> Gériatrie <input type="checkbox"/> Programmation logicielle <input type="checkbox"/> Analyse statistique & forage de données <input type="checkbox"/> Télématicque / Réseautique <input type="checkbox"/> Neurologie <input type="checkbox"/> Psychologie <input type="checkbox"/> Autre ... <i>⟨saisie libre⟩</i>

— suite à la page suivante —

Formulaire d'utilisation de CoLAB – suite

Code	Question	Format de réponse
Partie I : retour sur utilisation		
Votre cadre de travail		
SI.Q01	★ J'ai utilisé CoLab pour ...	<input type="checkbox"/> Construire une grille d'observation <input type="checkbox"/> Appliquer une grille d'observation <input type="checkbox"/> Rédiger un verbatim des participants <input type="checkbox"/> Valider les données des capteurs en les croisant avec les prises vidéo <input type="checkbox"/> Interpréter les données des capteurs en les croisant avec les prises vidéo <input type="checkbox"/> Autre ... <i>⟨saisie libre⟩</i>
SI.Q02	★ En rapport à la question précédente, comment jugeriez-vous l'apport de CoLab à votre travail ?	Nécessaire ○○○○○○○ Gadget
SI.Q03	★ Comment justifieriez-vous votre réponse à la question précédente ?	<i>⟨saisie libre⟩</i>
SI.Q04	★ À quel point êtes-vous d'accord avec cette phrase : « CoLab m'a aidé à collaborer avec les autres chercheurs du projet DEI. »	Pas du tout d'accord ○○○○○○○ Tout à fait d'accord
SI.Q05	★ Comment justifieriez-vous votre réponse à la question précédente ?	<i>⟨saisie libre⟩</i>
Bugs		
SI.Q06	★ Avez-vous rencontré beaucoup de bugs dans CoLab ?	Non, aucun ○○○○○○○ Oui, beaucoup

— suite à la page suivante —

Formulaire d'utilisation de COLAB – suite

Code	Question	Format de réponse
SI.Q07	Si vous avez rencontré des bugs, pouvez-vous en faire une brève description ?	<i>⟨saisie libre⟩</i>
SI.Q08	★ Les bugs rencontrés vous ont-ils empêché d'être productif dans votre tâche ?	Non, pas du tout ○○○○○○○ Oui, beaucoup
Fonctionnalités		
SI.Q09	★ [Sélecteur de source] Avez-vous utilisé les touches de clavier 1 à 6 pour changer de source ?	<input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'avais pas compris qu'on pouvait changer de point de capture avec les touches 1 à 6
SI.Q10	★ [Sélecteur de source] Avez-vous utilisé les info-bulles détaillant le status des vidéos, en survolant les icônes ?	<input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'avais pas remarqué qu'on pouvait survoler ces icônes
SI.Q11	★ [Sélecteur de source] En regard de votre travail, comment évalueriez vous l'utilité de pouvoir changer de source vidéo ?	Pas du tout utile ○○○○○○○ Très utile
SI.Q12	★ [Afficheur vidéo] Vous avez trouvé le chargement des vidéos ...	Très lent ○○○○○○○ Très rapide

— suite à la page suivante —

Formulaire d'utilisation de COLAB – suite

Code	Question	Format de réponse
SI.Q13	[Afficheur vidéo] Avez-vous trouvé le positionnement des caméras pertinent (exhaustif)? Si non, merci de préciser les informations qui vous ont manqué ou les suggestions que vous apporteriez.	<i>⟨saisie libre⟩</i>
SI.Q14	★ [Barre de contrôle] Avez-vous utilisé le raccourci clavier "barre d'espace" pour la marche / pause de la vidéo?	<input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je ne savais pas qu'on pouvait mettre en marche / pause la vidéo avec la barre d'espace
SI.Q15	★ [Barre de contrôle] Avez-vous utilisé le bouton pour accélérer / ralentir la vitesse de lecture des vidéos?	<input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'y ai pas prêté attention
SI.Q16	★ [Barre de contrôle] Avez-vous utilisé le bouton pour masquer / afficher l'explorateur temporel?	<input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'y ai pas prêté attention
SI.Q17	★ [Explorateur temporel] J'ai trouvé le fait de pouvoir accéder aux tâches ...	Pas du tout utile ○○○○○○○ Très utile
SI.Q18	★ [Explorateur temporel] J'ai trouvé l'utilisation de l'explorateur temporel...	Pas du tout commode ○○○○○○○ Très commode

— suite à la page suivante —

Formulaire d'utilisation de COLAB – suite

Code	Question	Format de réponse
SI.Q19	★ [Explorateur temporel] J'ai trouvé les infobulles	<div>Pas du tout utiles ○○○○○○○○ Très utiles</div>
SI.Q20	★ [Explorateur temporel] J'ai trouvé les infobulles	<div>Pas du tout faciles à comprendre ○○○○○○○○ Très faciles à comprendre</div>
SI.Q21	★ [Explorateur temporel] Avez-vous utilisé le bouton servant à démarrer la vidéo à partir du début d'un segment ?	<div> <input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'y ai pas prêté attention </div>
SI.Q22	[Explorateur temporel] Avez-vous des remarques ou suggestions à faire sur l'explorateur temporel ?	<div>⟨saisie libre⟩</div>
SI.Q23	★ [Barre d'informations] Avez-vous consulté les icônes pour vous renseigner sur le statut de l'expérimentation ?	<div> <input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'y avais pas prêté attention </div>
SI.Q24	★ [Barre d'informations] Avez-vous utilisé le bouton d'extension / réduction du menu ?	<div> <input type="radio"/> Oui, et ça m'a été très utile <input type="radio"/> Oui, et ça m'a été plutôt utile <input type="radio"/> Oui, mais ça ne m'a pas été utile <input type="radio"/> Non, je n'en voyais pas en quoi ça pouvait m'être utile <input type="radio"/> Non, je n'y avais pas prêté attention </div>

— suite à la page suivante —

Formulaire d'utilisation de CoLAB – *suite*

Code	Question	Format de réponse	
Partie II : appréciation subjective			
SII.Q01	★ Je trouve CoLab plutôt	Humain	○ ○ ○ ○ ○ ○ ○ Technique
SII.Q02	★ Je pense que CoLab	M'isole	○ ○ ○ ○ ○ ○ ○ Me sociabilise
SII.Q03	★ Je trouve CoLab plutôt	Plaisant	○ ○ ○ ○ ○ ○ ○ Déplaisant
SII.Q04	★ Je trouve CoLab plutôt	Original	○ ○ ○ ○ ○ ○ ○ Conventionnel
SII.Q05	★ Je trouve CoLab plutôt	Simple	○ ○ ○ ○ ○ ○ ○ Compliqué
SII.Q06	★ Je trouve CoLab plutôt	Professionnel	○ ○ ○ ○ ○ ○ ○ Amateur
SII.Q07	★ Je trouve CoLab plutôt	Laid	○ ○ ○ ○ ○ ○ ○ Beau
SII.Q08	★ Je trouve CoLab plutôt	Pratique	○ ○ ○ ○ ○ ○ ○ Pas pratique
SII.Q09	★ Je trouve CoLab plutôt	Agréable	○ ○ ○ ○ ○ ○ ○ Désagréable

— suite à la page suivante —

Formulaire d'utilisation de CoLAB – suite

Code	Question	Format de réponse		
SII.Q10	★ Je trouve CoLab plutôt	Fastidieux	○ ○ ○ ○ ○ ○ ○	Efficace
SII.Q11	★ Je trouve CoLab plutôt	De bon goût	○ ○ ○ ○ ○ ○ ○	De mauvais goût
SII.Q12	★ Je trouve CoLab plutôt	Prévisible	○ ○ ○ ○ ○ ○ ○	Imprévisible
SII.Q13	★ Je trouve CoLab plutôt	Bas de gamme	○ ○ ○ ○ ○ ○ ○	Haut de gamme
SII.Q14	★ Je trouve que CoLab	M'exclut	○ ○ ○ ○ ○ ○ ○	M'intègre
SII.Q15	★ Je trouve que CoLab	Me rapproche des autres	○ ○ ○ ○ ○ ○ ○	Me sépare des autres
SII.Q16	★ Je trouve CoLab plutôt	Non présentable	○ ○ ○ ○ ○ ○ ○	Présentable
SII.Q17	★ Je trouve CoLab plutôt	Rebutant	○ ○ ○ ○ ○ ○ ○	Attirant
SII.Q18	★ Je trouve CoLab plutôt	Sans imagination	○ ○ ○ ○ ○ ○ ○	Créatif
SII.Q19	★ Je trouve CoLab plutôt	Bon	○ ○ ○ ○ ○ ○ ○	Mauvais

— suite à la page suivante —

Formulaire d'utilisation de CoLAB – suite

Code	Question	Format de réponse		
SII.Q20	★ Je trouve CoLab plutôt	Confus	○ ○ ○ ○ ○ ○ ○	Clair
SII.Q21	★ Je trouve CoLab plutôt	Repoussant	○ ○ ○ ○ ○ ○ ○	Attrayant
SII.Q22	★ Je trouve CoLab plutôt	Audacieux	○ ○ ○ ○ ○ ○ ○	Prudent
SII.Q23	★ Je trouve CoLab plutôt	Novateur	○ ○ ○ ○ ○ ○ ○	Conservateur
SII.Q24	★ Je trouve CoLab plutôt	Ennuyeux	○ ○ ○ ○ ○ ○ ○	Captivant
SII.Q25	★ Je trouve CoLab plutôt	Peu exigeant	○ ○ ○ ○ ○ ○ ○	Challenging
SII.Q26	★ Je trouve CoLab plutôt	Motivant	○ ○ ○ ○ ○ ○ ○	Décourageant
SII.Q27	★ Je trouve CoLab plutôt	Incontrôlable	○ ○ ○ ○ ○ ○ ○	Maîtrisable
SII.Q28	★ Je trouve CoLab plutôt	Nouveau	○ ○ ○ ○ ○ ○ ○	Commun
SII.Q29	★ Je pense que mes réponses à cette section étaient	Sincères	○ ○ ○ ○ ○ ○ ○	Complaisantes

— suite à la page suivante —

Code	Question	Format de réponse
Partie III : CoLab, positionnement et évolutions possibles		
Positionnement		
SIIL.Q01	★ Avez-vous déjà utilisé des outils similaires à Co-Lab ? Si oui, lesquels ? Quelles étaient leurs points forts et leurs points faibles par rapport à CoLab ?	⟨saisie libre⟩
SIIL.Q02	★ Quels sont, d'après vous, les points forts de Co-Lab ?	⟨saisie libre⟩
SIIL.Q03	★ Quels sont, d'après vous, les points faibles de Co-Lab ?	⟨saisie libre⟩
SIIL.Q04	★ Quelles améliorations et nouveautés dans CoLab vous donneraient plus envie de l'utiliser ?	⟨saisie libre⟩
L'introduction des verbatim Par verbatim, nous entendons « la retranscription textuelle de paroles, et éventuellement d'actes, de gestes et d'expressions corporelles, constituée d'un ensemble d'observations datées dans le temps. »		
SIIL.Q06	★ Êtes vous d'accord avec cette définition ?	<input type="radio"/> Oui <input type="radio"/> Non <input type="radio"/> Pas d'avis
SIIL.Q07	Quelles nuances apporteriez-vous à la définition de « verbatim », si vous n'êtes pas totalement d'accord avec notre définition ?	⟨saisie libre⟩
SIIL.Q08	[Verbatim] Pour moi, la possibilité de consulter et d'éditer des observations de verbatim serait ...	Pas du tout utile ○○○○○○○ Très utile

— suite à la page suivante —

Formulaire d'utilisation de COLAB – suite

Code	Question	Format de réponse		
SIIL.Q09	[Verbatim] Pour moi, la possibilité de sous-titrer les vidéos avec les observations de verbatim serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q10	[Verbatim] Pour moi, la possibilité de lancer la vidéo quelques secondes avant le début d'une observation de verbatim serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile
Vidéos				
SIIL.Q11	[Vidéos] Pour moi, la possibilité qu'on m'indique, dans l'explorateur temporel et pour chaque source vidéo, les moments où du mouvement est détecté serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q12	[Vidéos] Pour moi, la possibilité de voir l'emplacement des sources vidéo sur une carte 2D de l'appartement serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q13	[Vidéo] Pour moi, la possibilité de voir plusieurs sources (vidéo) en même temps, sous la forme d'une mosaïque serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q14	[Vidéo] Pour moi, la possibilité de voir des vignettes lorsque je survole la barre d'avancement du lecteur vidéo avec mon curseur serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○	Très utile

— suite à la page suivante —

Formulaire d'utilisation de COLAB – suite

Code	Question	Format de réponse
<p>Grilles d'évaluation</p> <p>Nous définissons une grille d'évaluation comme « un ensemble de critères structurés sous forme d'arbre dont les feuilles regroupent des ensembles d'observations, et dont les nœuds de niveaux inférieur constituent des catégories d'autant plus générales qu'elles sont proches de la racine. »</p> <p>Dans notre définition, une observation de grille est un fait rapporté qui participe au remplissage de la grille, lorsque la grille est déjà saturée, et qu'on évalue un participant. Par exemple, le critère 'Lecture à voix haute des consignes' de la grille appliquée au participant n°15 est nourri par trois observations de grille, respectivement aux 12, 15 et 22e minutes.</p> <p>Une observation de grille est donc toujours rattachée à un critère de grille, à un participant et elle est localisée dans le temps.</p>		
SIII.Q15	★ Êtes vous d'accord avec cette définition ?	<input type="radio"/> Oui <input type="radio"/> Non <input type="radio"/> Pas d'avis
SIII.Q16	Si vous voulez apporter une nuance ou une objection à notre définition de grille d'évaluation ...	<i>⟨saisie libre⟩</i>
SIII.Q17	★ Êtes vous d'accord avec l'affirmation suivante : « Une observation de grille devrait toujours correspondre à une observation de verbatim. »	<input type="radio"/> Oui <input type="radio"/> Non <input type="radio"/> Je ne sais pas
SIII.Q18	[Grille d'évaluation] Pour moi, la possibilité d'associer facilement une observation de verbatim avec une observation de grille serait ...	Pas du tout utile ○○○○○○○ Très utile
SIII.Q19	[Grille d'évaluation] Pour moi, la possibilité d'appliquer la grille en mettant en place la fidélité inter-juge avec un binôme de travail serait ...	Pas du tout utile ○○○○○○○ Très utile
SIII.Q20	[Grille d'évaluation] Pour moi, la possibilité d'appliquer la grille en mettant en place la fidélité inter-juge entre 3 (et plus) collaborateurs serait ...	Pas du tout utile ○○○○○○○ Très utile

— suite à la page suivante —

Formulaire d'utilisation de CoLAB – suite

Code	Question	Format de réponse		
SIIL.Q21	[Grille d'évaluation] Pour moi, la possibilité d'utiliser un temps d'exécution d'une tâche comme un critère de grille serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q22	[Grille d'évaluation] Pour moi, la possibilité d'alimenter, collaborativement, les critères et catégories d'une grille d'évaluation serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q23	[Grille d'évaluation] Pour moi, la possibilité de figer la grille (c'est-à-dire la rendre impossible à modifier) lorsqu'elle est saturée serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○ ○	Très utile
Capteurs				
SIIL.Q24	[Capteurs] Pour moi, la possibilité de faire apparaître dans l'explorateur temporel des événements d'un ensemble de capteurs que j'ai définis serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○ ○	Très utile
SIIL.Q25	[Capteurs] Pour moi, la possibilité de voir l'emplacement de ces capteurs sur une carte 2D de l'appartement serait ...	Pas du tout utile	○ ○ ○ ○ ○ ○ ○ ○	Très utile
Commentaires				
SIIL.Q26	Auriez-vous d'autres suggestions, commentaires ou appréciations à faire sur CoLab ?	⟨saisie libre⟩		

F.2 Réponses au questionnaire

Les réponses de type échelle de Likert sont retranscrites sur une échelle de -3 à $+3$. Les valeurs de ces réponses sont écrites avec la syntaxe suivante : $X / \pm 3$ avec X la valeur sur l'échelle de Likert. Les valeurs associées aux deux pôles de l'échelle sont données dans la question, avec la syntaxe suivante : [*Terme 1* – *Terme 2*]. Une valeur de -3 correspond à une égalité parfaite avec le *Terme 1* et réciproquement, une valeur de $+3$ correspond à une égalité parfaite avec le *Terme 2*.

Le questionnaire a été complété par quatre utilisateurs sur les six ayant essayé le logiciel. Les deux manquants n'ont pas estimé avoir assez testé le logiciel pour être capable de renseigner le questionnaire. Les données de ces utilisateurs ont été anonymisées et retranscrites dans les colonnes *U1*, *U2*, *U3* et *U4*.

TABLEAU F.2 – Réponses au formulaire d'utilisation de CoLAB

Code	Question	U1	U2	U3	U4
Préambule					
Renseignements personnels					
P.Q01	★ Autorisez-vous qu'on reprenne contact avec vous pour obtenir des précisions sur votre perception de l'outil ?	Oui	Oui	Oui	Oui
P.Q02	Quel est votre âge ?	24	27	34	30
Mise en contexte					
P.Q03	★ Comment évalueriez-vous votre confort avec les outils numériques ? [<i>Pas du tout confortable – Très confortable</i>]	2 / ± 3	3 / ± 3	2 / ± 3	3 / ± 3
P.Q04	★ Mon / mes domaine(s) d'expertise sont ...	Ergothérapie	Programmation logicielle, Analyse statistique & forage de données, Télématique / Réseautique	Ergothérapie, Gériatrie	Gériatrie, Analyse statistique & forage de données, psychologie
— suite à la page suivante —					

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
Partie I : retour sur utilisation					
Votre cadre de travail					
SI.Q01	★ J'ai utilisé CoLab pour ...	Construire une grille..., appliquer une grille...	Valider les données des capteurs..., interpréter les données des capteurs	Rédiger un verbatim des participants	Construire une grille..., appliquer une grille..., valider les données des capteurs..., interpréter les données des capteurs..., rédiger un verbatim des participants
SI.Q02	★ En rapport à la question précédente, comment jugeriez-vous l'apport de CoLab à votre travail ? [Nécessaire – Gadget]	-2 / ± 3	-2 / ± 3	-3 / ± 3	-2 / ± 3
SI.Q03	★ Comment justifieriez-vous votre réponse à la question précédente ?	A permis de recueillir des données sur plusieurs participants	Très utile pour valider les données des capteurs, mais pas absolument nécessaire si on n'avait pas eu tant de problèmes ou encore si on n'avait pas de vidéo.	l'idée initiale d'avoir plusieurs caméra pour filmer une tâche n'était pas super, car ça demandait ensuite d'étudier plusieurs vidéos en parallèle pour avoir une vue d'ensemble. CoLab a toutefois permis de répondre à cette problématique	Je pense que c'est beaucoup de travail d'annoter les activités correctement sur la barre du temps, mais on gagne beaucoup de temps par la suite si c'est un projet sur lequel on est nombreux et si on pense réutiliser beaucoup ces données.
SI.Q04	★ À quel point êtes-vous d'accord avec cette phrase : « CoLab m'a aidé à collaborer avec les autres chercheurs du projet DEI. » [Pas du tout d'accord – Tout à fait d'accord]	0 / ± 3	0 / ± 3	0 / ± 3	2 / ± 3
— suite à la page suivante —					

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
SI.Q05	★ Comment justifieriez-vous votre réponse à la question précédente ?	Je me suis surtout servi des vidéos, la collaboration avec d'autres était moins le but de mon projet.	Je n'ai pas vraiment utilisé CoLab pour collaborer avec d'autres chercheurs, excepté ceux du Domus peut-être et XXXX.	Je n'ai pas eu à collaborer avec d'autres chercheurs.	Partager sécurisé des vidéos en ligne (et non échange de disque dur physique)
Bugs					
SI.Q06	★ Avez-vous rencontré beaucoup de bugs dans CoLab ? [<i>Non, aucun – Oui, beaucoup</i>]	0 / ± 3	-2 / ± 3	-3 / ± 3	-2 / ± 3
SI.Q07	Si vous avez rencontré des bugs, pouvez-vous en faire une brève description ?	Son qui ne fonctionne pas dans toutes les prises de vue, caméras qui ne fonctionnent pas dans toutes les pièces, image qui prend du temps à charger lorsque l'on change de caméra, une vidéo que j'ai dû ouvrir trois fois avant qu'elle fonctionne	-	-	Vidéos non synchronisée (pas vraiment un bug du colab comme tel par contre)
SI.Q08	★ Les bugs rencontrés vous ont-ils empêché d'être productif dans votre tâche ? [<i>Non, pas du tout – Oui, beaucoup</i>]	0 / ± 3	0 / ± 3	-3 / ± 3	-2 / ± 3
Fonctionnalités					
SI.Q09	★ [Sélecteur de source] Avez-vous utilisé les touches de clavier 1 à 6 pour changer de source ?	Non, je n'avais pas compris...	Non, je n'avais pas compris...	Oui, et ça m'a été très utile	Oui, et ça m'a été très utile
— suite à la page suivante —					

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SI.Q10	★ [Sélecteur de source] Avez-vous utilisé les infobulles détaillant le status des vidéos, en survolant les icônes ?	Oui, et ça m'a été plutôt utile	Oui, et ça m'a été très utile	Oui, et ça m'a été plutôt utile	Oui, et ça m'a été très utile
SI.Q11	★ [Sélecteur de source] En regard de votre travail, comment évalueriez vous l'utilité de pouvoir changer de source vidéo ? [Pas du tout utile – Très utile]	3 / ± 3	3 / ± 3	2 / ± 3	1 / ± 3
SI.Q12	★ [Afficheur vidéo] Vous avez trouvé le chargement des vidéos ... [Très lent – Très rapide]	0 / ± 3	2 / ± 3	3 / ± 3	3 / ± 3
SI.Q13	[Afficheur vidéo] Avez-vous trouvé le positionnement des caméras pertinent (exhaustif) ? Si non, merci de préciser les informations qui vous ont manqué ou les suggestions que vous apporteriez.	Oui pertinent, mais parfois nous n'avions pas tous les angles de vue pour bien suivre le participants, il aurait fallu plus de capteurs (ex. salle de bain, chambre)	Il manque une vue de l'entrée (porte).	J'ai trouvé le fait d'avoir plusieurs caméra pour filmer la tâche non pertinent. De plus, les caméra étaient fixes et parfois, le sujet faisait quelque chose hors champ. J'ai travaillé avec plus d'une 50e de vidéos où il y avait quelqu'un qui filmait, et j'ai trouvé cette expérience plus facile et plus riche en information de façon générale. Je pense que même pour ce genre de vidéos (1 seule vidéo), CoLab pourrait apporter des choses vraiment intéressantes, comme le fait d'identifier les tâches dans le temps.	Oui sinon que j'aurais aimé un caméra au dessus du combiné de téléphone pour voir les numéros composé et ce qui est fouillé dans le bottin

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SI.Q14	★ [Barre de contrôle] Avez-vous utilisé le raccourci clavier "barre d'espace" pour la marche / pause de la vidéo ?	Oui, et ça m'a été très utile	Oui, et ça m'a été très utile	Oui, et ça m'a été très utile	Oui, et ça m'a été très utile
SI.Q15	★ [Barre de contrôle] Avez-vous utilisé le bouton pour accélérer / ralentir la vitesse de lecture des vidéos ?	Oui, et ça m'a été très utile	Oui, et ça m'a été très utile	Oui, et ça m'a été plutôt utile	Oui, et ça m'a été très utile
SI.Q16	★ [Barre de contrôle] Avez-vous utilisé le bouton pour masquer / afficher l'explorateur temporel ?	Oui, mais ça ne m'a pas été utile	Oui, et ça m'a été plutôt utile	Non, je n'y ai pas prêté attention	Non, je n'y ai pas prêté attention
SI.Q17	★ [Explorateur temporel] J'ai trouvé le fait de pouvoir accéder aux tâches ... [<i>Pas du tout utile – Très utile</i>]	2 / ± 3	2 / ± 3	3 / ± 3	3 / ± 3
SI.Q18	★ [Explorateur temporel] J'ai trouvé l'utilisation de l'explorateur temporel... [<i>Pas du tout commode – Très commode</i>]	2 / ± 3	2 / ± 3	3 / ± 3	2 / ± 3
SI.Q19	★ [Explorateur temporel] J'ai trouvé les infobulles [<i>Pas du tout utiles – Très utiles</i>]	-1 / ± 3	2 / ± 3	3 / ± 3	2 / ± 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – *suite*

Code	Question	U1	U2	U3	U4
SI.Q20	★ [Explorateur temporel] J'ai trouvé les infobulles <i>[Pas du tout faciles à comprendre – Très faciles à comprendre]</i>	3 / ± 3	2 / ± 3	3 / ± 3	2 / ± 3
SI.Q21	★ [Explorateur temporel] Avez-vous utilisé le bouton servant à démarrer la vidéo à partir du début d'un segment ?	Non, je n'y ai pas prêté attention	Oui, et ça m'a été plutôt utile.	Non, je n'y ai pas prêté attention	Non, je n'y ai pas prêté attention
SI.Q22	[Explorateur temporel] Avez-vous des remarques ou suggestions à faire sur l'explorateur temporel ?	-	-	-	Oui ! Souvent je voulais retourner à un temps que j'avais noté (ex : 03 :45), et j'aurais aimé pouvoir glisser la ligne sur la barre du temps et voir le temps que ça me donnerait si je la lâchais à cette endroit plutôt que de cliquer approximativement et charger la vidéo pour rien.
SI.Q23	★ [Barre d'informations] Avez-vous consulté les icônes pour vous renseigner sur le statut de l'expérimentation ?	Oui, mais ça ne m'a pas été utile	Oui, et ça m'a été très utile	Non, je n'y avais pas prêté attention	Oui, et ça m'a été plutôt utile
SI.Q24	★ [Barre d'informations] Avez-vous utilisé le bouton d'extension / réduction du menu ?	Oui, mais ça ne m'a pas été utile	Oui, et ça m'a été plutôt utile	Non, je n'y avais pas prêté attention	Non, je n'y avais pas prêté attention

— suite à la page suivante —

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
Partie II : appréciation subjective					
SII.Q01	★ Je trouve CoLab plutôt [<i>Humain – technique</i>]	1 / ± 3	-1 / ± 3	0 / ± 3	0 / ± 3
SII.Q02	★ Je pense que CoLab [<i>M'isole – Me sociabilise</i>]	0 / ± 3	1 / ± 3	0 / ± 3	2 / ± 3
SII.Q03	★ Je trouve CoLab plutôt [<i>Plaisant – Déplaisant</i>]	-1 / ± 3	-3 / ± 3	-2 / ± 3	-2 / ± 3
SII.Q04	★ Je trouve CoLab plutôt [<i>Original – Conventionnel</i>]	-1 / ± 3	-2 / ± 3	-2 / ± 3	-1 / ± 3
SII.Q05	★ Je trouve CoLab plutôt [<i>Simple – Compliqué</i>]	-2 / ± 3	1 / ± 3	-2 / ± 3	-2 / ± 3
SII.Q06	★ Je trouve CoLab plutôt [<i>Professionnel – Amateur</i>]	-1 / ± 3	-3 / ± 3	-2 / ± 3	-2 / ± 3
SII.Q07	★ Je trouve CoLab plutôt [<i>Laid – Beau</i>]	1 / ± 3	2 / ± 3	1 / ± 3	2 / ± 3
SII.Q08	★ Je trouve CoLab plutôt [<i>Pratique – Pas pratique</i>]	-2 / ± 3	-3 / ± 3	-2 / ± 3	-2 / ± 3
SII.Q09	★ Je trouve CoLab plutôt [<i>Agréable – Désagréable</i>]	0 / ± 3	-2 / ± 3	-2 / ± 3	-2 / ± 3
SII.Q10	★ Je trouve CoLab plutôt [<i>Fastidieux – Efficace</i>]	-1 / ± 3	2 / ± 3	2 / ± 3	2 / ± 3
— suite à la page suivante —					

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
SII.Q11	★ Je trouve CoLab plutôt [<i>De bon goût – De mauvais goût</i>]	-2 / ± 3	-2 / ± 3	-2 / ± 3	-3 / ± 3
SII.Q12	★ Je trouve CoLab plutôt [<i>Prévisible – Imprévisible</i>]	-2 / ± 3	-3 / ± 3	-3 / ± 3	-2 / ± 3
SII.Q13	★ Je trouve CoLab plutôt [<i>Bas de gamme – Haut de gamme</i>]	1 / ± 3	2 / ± 3	2 / ± 3	2 / ± 3
SII.Q14	★ Je trouve que CoLab [<i>M'exclut – M'intègre</i>]	1 / ± 3	1 / ± 3	0 / ± 3	1 / ± 3
SII.Q15	★ Je trouve que CoLab [<i>Me rapproche des autres – Me sépare des autres</i>]	0 / ± 3	-1 / ± 3	0 / ± 3	-1 / ± 3
SII.Q16	★ Je trouve CoLab plutôt [<i>Non présentable – Présentable</i>]	2 / ± 3	3 / ± 3	2 / ± 3	3 / ± 3
SII.Q17	★ Je trouve CoLab plutôt [<i>Rebutant – Attirant</i>]	1 / ± 3	2 / ± 3	2 / ± 3	2 / ± 3
SII.Q18	★ Je trouve CoLab plutôt [<i>Sans imagination – Créatif</i>]	1 / ± 3	3 / ± 3	2 / ± 3	1 / ± 3
SII.Q19	★ Je trouve CoLab plutôt [<i>Bon – Mauvais</i>]	-2 / ± 3	-3 / ± 3	-2 / ± 3	-3 / ± 3
SII.Q20	★ Je trouve CoLab plutôt [<i>Confus – Clair</i>]	1 / ± 3	1 / ± 3	2 / ± 3	2 / ± 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
SII.Q21	★ Je trouve CoLab plutôt [<i>Repoussant – Attrayant</i>]	1 / ± 3	2 / ± 3	2 / ± 3	2 / ± 3
SII.Q22	★ Je trouve CoLab plutôt [<i>Audacieux – Prudent</i>]	-1 / ± 3	-3 / ± 3	-1 / ± 3	0 / ± 3
SII.Q23	★ Je trouve CoLab plutôt [<i>Novateur – Conservateur</i>]	-1 / ± 3	-3 / ± 3	-1 / ± 3	-1 / ± 3
SII.Q24	★ Je trouve CoLab plutôt [<i>Ennuyeux – Captivant</i>]	0 / ± 3	2 / ± 3	1 / ± 3	0 / ± 3
SII.Q25	★ Je trouve CoLab plutôt [<i>Peu exigeant – Challenging</i>]	1 / ± 3	0 / ± 3	-1 / ± 3	0 / ± 3
SII.Q26	★ Je trouve CoLab plutôt [<i>Motivant – Décourageant</i>]	0 / ± 3	-2 / ± 3	-2 / ± 3	-1 / ± 3
SII.Q27	★ Je trouve CoLab plutôt [<i>Incontrôlable – Maîtrisable</i>]	2 / ± 3	2 / ± 3	3 / ± 3	2 / ± 3
SII.Q28	★ Je trouve CoLab plutôt [<i>Nouveau – Commun</i>]	1 / ± 3	1 / ± 3	2 / ± 3	0 / ± 3
SII.Q29	★ Je pense que mes réponses à cette section étaient [<i>Sincères – Complaisantes</i>]	-3 / ± 3	-2 / ± 3	-3 / ± 3	-3 / ± 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
Partie III : CoLab, positionnement et évolutions possibles					
Positionnement					
SIII.Q01	★ Avez-vous déjà utilisé des outils similaires à CoLab ? Si oui, lesquels ? Quelles étaient leurs point forts et leurs points faibles par rapport à CoLab ?	Je n'ai jamais vraiment utilisé d'outils similaires. La seule comparaison que je puisse faire est avec la méthode utilisée pour filmer les participants à Montréal, soit via une caméra tenue par un observateur.	Logiciels d'édition vidéo ou audio tels que Windows Movie Maker, Cool Edit Pro / Adobe Audition, Audacity, Blender. On y retrouve surtout une ressemblance du côté lecture du vidéo/audio et de l'explorateur temporel.	Non.	Non.
SIII.Q02	★ Quels sont, d'après vous, les points forts de CoLab ?	CoLab permet à l'observateur de choisir lui-même ce qu'il souhaite observer dans la séquence vidéo. Beaucoup de points de vue différents sont disponibles dans la cuisine, ce qui est bien.	WebApp accessible partout, tout le nécessaire pour une EXP dans une seule page ("single page webapp" qu'on dit ???), raccourcis claviers, interface conviviale et sobre, divers panneaux sont rétractables (i.e. explorateur temporel) ou multi-fonctions (i.e. celui de gauche). Pis surtout : ça fait la job !	Pouvoir identifier des moments précis dans une vidéo	Multi-caméra, annotation, fast-forward, en ligne, esthétique et simplicité
— suite à la page suivante —					

Réponses au formulaire d'utilisation de CoLAB – suite

Code	Question	U1	U2	U3	U4
SIIL.Q03	★ Quels sont, d'après vous, les points faibles de CoLab ?	Le fait que les capteurs ne fonctionnaient pas toujours ou ne transmettaient pas le son. Comme ils ne sont pas mobiles, parfois ils ne permettent pas de bien suivre le participant dans une pièce, et on ne peut pas zoomer. Le fait qu'il y ait un délai de quelque secondes chaque fois que l'on change de point de vue rend cela un peu fastidieux (long). Le fait que le son et l'image ne soient pas toujours synchronisés rend parfois le suivi de l'action difficile.	Je veux pas être complaisant, mais je n'en trouve pas vraiment. Peut-être juste que c'est hosté sur un serveur de l'université pas toujours fiable on dirait... (je ne peux accéder à CoLab en ce moment) mais sinon ça va.	possibilité de créer des signets pour des moments X, mais impossibilité de faire ressortir un rapport avec ces moments x. Aussi, créer des groupes de signets avec différentes étiquettes supplémentaires, puis ressortir tout ça en rapport (ex. signet erreur, étiquette correspondant au type d'erreur; puis rapport qui me sort qu'il y a eu 10 erreurs, dont 3 de telle nature et 7 de l'autre.)	Peut être time-consuming donc doit correspondre à 100% des besoins pour ça vaille la peine d'y dédier une équipe. Aussi ça prendrait des façons de brouiller les ID pour faire de l'évaluation à l'aveugle si nécessaire.
SIIL.Q04	★ Quelles améliorations et nouveautés dans CoLab vous donneraient plus envie de l'utiliser ?	Avoir un bouton pour reculer (pas juste avancer) les vidéos et l'amélioration des éléments mentionnés dans les points faibles.	Bouton pour recommencer le vidéo du début (ça m'aurait surtout été utile pour corriger les problèmes de synchro, mais utile quand même) Pouvoir regarder plusieurs vidéos en même temps ;-) Pouvoir voir l'état des capteurs en "temps réel" pendant un vidéo :-P	voir plus haut	Je pense qu'il serait super avantageux d'avoir une discussion avec Mélanie Couture qui se spécialise dans la prise de verbatim et l'analyse qualitative. Elle saurait mieux que moi comment le logiciel pourrait devenir indispensable dans ce genre de contexte.
L'introduction des verbatim Par verbatim, nous entendons « la retranscription textuelle de paroles, et éventuellement d'actes, de gestes et d'expressions corporelles, constituée d'un ensemble d'observations datées dans le temps. »					
SIIL.Q06	★ Êtes vous d'accord avec cette définition ?	Oui	Oui	Oui	Oui

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SIII.Q07	Quelles nuances apporteriez-vous à la définition de « verbatim », si vous n'êtes pas totalement d'accord avec notre définition ?	-	-	-	-
SIII.Q08	[Verbatim] Pour moi, la possibilité de consulter et d'éditer des observations de verbatim serait ... <i>[Pas du tout utile – Très utile]</i>	1 / \pm 3	2 / \pm 3	3 / \pm 3	3 / \pm 3
SIII.Q09	[Verbatim] Pour moi, la possibilité de sous-titrer les vidéos avec les observations de verbatim serait ...	1 / \pm 3	0 / \pm 3	2 / \pm 3	3 / \pm 3
SIII.Q10	[Verbatim] Pour moi, la possibilité de lancer la vidéo quelques secondes avant le début d'une observation de verbatim serait ... <i>[Pas du tout utile – Très utile]</i>	1 / \pm 3	2 / \pm 3	1 / \pm 3	2 / \pm 3
Vidéos					
SIII.Q11	[Vidéos] Pour moi, la possibilité qu'on m'indique, dans l'explorateur temporel et pour chaque source vidéo, les moments où du mouvement est détecté serait ... <i>[Pas du tout utile – Très utile]</i>	1 / \pm 3	3 / \pm 3	3 / \pm 3	3 / \pm 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SIIL.Q12	[Vidéos] Pour moi, la possibilité de voir l'emplacement des sources vidéo sur une carte 2D de l'appartement serait ... [<i>Pas du tout utile – Très utile</i>]	1 / \pm 3	3 / \pm 3	1 / \pm 3	1 / \pm 3
SIIL.Q13	[Vidéo] Pour moi, la possibilité de voir plusieurs sources (vidéo) en même temps, sous la forme d'une mosaïque serait ... [<i>Pas du tout utile – Très utile</i>]	1 / \pm 3	2 / \pm 3	1 / \pm 3	1 / \pm 3
SIIL.Q14	[Vidéo] Pour moi, la possibilité de voir des vignettes lorsque je survole la barre d'avancement du lecteur vidéo avec mon curseur serait ... [<i>Pas du tout utile – Très utile</i>]	2 / \pm 3	3 / \pm 3	2 / \pm 3	0 / \pm 3
<p>Grilles d'évaluation</p> <p>Nous définissons une grille d'évaluation comme « un ensemble de critères structurés sous forme d'arbre dont les feuilles regroupent des ensembles d'observations, et dont les nœuds de niveaux inférieur constituent des catégories d'autant plus générales qu'elles sont proches de la racine. »</p> <p>Dans notre définition, une observation de grille est un fait rapporté qui participe au remplissage de la grille, lorsque la grille est déjà saturée, et qu'on évalue un participant. Par exemple, le critère 'Lecture à voix haute des consignes' de la grille appliquée au participant n°15 est nourri par trois observations de grille, respectivement aux 12, 15 et 22e minutes.</p> <p>Une observation de grille est donc toujours rattachée à un critère de grille, à un participant et elle est localisée dans le temps.</p>					
SIIL.Q15	★ Êtes vous d'accord avec cette définition ?	Oui	Pas d'avis	Oui	Oui

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SIII.Q16	Si vous voulez apporter une nuance ou une objection à notre définition de grille d'évaluation ...	-	-	-	-
SIII.Q17	★ Êtes vous d'accord avec l'affirmation suivante : « Une observation de grille devrait toujours correspondre à une observation de verbatim. »	Non	Je ne sais pas	Oui	Je ne sais pas
SIII.Q18	[Grille d'évaluation] Pour moi, la possibilité d'associer facilement une observation de verbatim avec une observation de grille serait ... [<i>Pas du tout utile – Très utile</i>]	2 / ± 3	0 / ± 3	3 / ± 3	3 / ± 3
SIII.Q19	[Grille d'évaluation] Pour moi, la possibilité d'appliquer la grille en mettant en place la fidélité inter-juge avec un binôme de travail serait ... [<i>Pas du tout utile – Très utile</i>]	2 / ± 3	0 / ± 3	3 / ± 3	3 / ± 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
SIII.Q20	[Grille d'évaluation] Pour moi, la possibilité d'appliquer la grille en mettant en place la fidélité inter-juge entre 3 (et plus) collaborateurs serait ... [<i>Pas du tout utile – Très utile</i>]	2 / \pm 3	0 / \pm 3	3 / \pm 3	2 / \pm 3
SIII.Q21	[Grille d'évaluation] Pour moi, la possibilité d'utiliser un temps d'exécution d'une tâche comme un critère de grille serait ... [Pas du tout utile – Très utile]	1 / \pm 3	0 / \pm 3	3 / \pm 3	1 / \pm 3
SIII.Q22	[Grille d'évaluation] Pour moi, la possibilité d'alimenter, collaborativement, les critères et catégories d'une grille d'évaluation serait ... [Pas du tout utile – Très utile]	2 / \pm 3	0 / \pm 3	3 / \pm 3	1 / \pm 3
SIII.Q23	[Grille d'évaluation] Pour moi, la possibilité de figer la grille (c'est-à-dire la rendre impossible à modifier) lorsqu'elle est saturée serait ... [Pas du tout utile – Très utile]	2 / \pm 3	0 / \pm 3	0 / \pm 3	1 / \pm 3

— suite à la page suivante —

Réponses au formulaire d'utilisation de COLAB – suite

Code	Question	U1	U2	U3	U4
Capteurs					
SIIL.Q24	[Capteurs] Pour moi, la possibilité de faire apparaître dans l'explorateur temporel des événements d'un ensemble de capteurs que j'ai définis serait ... [Pas du tout utile – Très utile]	2 / \pm 3	3 / \pm 3	3 / \pm 3	3 / \pm 3
SIIL.Q25	[Capteurs] Pour moi, la possibilité de voir l'emplacement de ces capteurs sur une carte 2D de l'appartement serait ... [Pas du tout utile – Très utile]	2 / \pm 3	3 / \pm 3	2 / \pm 3	0 / \pm 3
Commentaires					
SIIL.Q26	Auriez-vous d'autres suggestions, commentaires ou appréciations à faire sur CoLab ?	Juste pour clarification pour les deux questions qui suggèrent une vue en plan des caméras et capteurs : ces fonctionnalités seraient à mon avis très utiles pour quelqu'un qui ne connaîtrait pas déjà tout ça par cœur comme moi, donc n'importe qui d'autre dans un labo qui n'a pas passé son temps à se battre avec l'infra du DOMUS.	-	-	-

Bibliographie

- [1] AIKEN, M., *The Cyber Effect : A Pioneering Cyberpsychologist Explains How Human Behavior Changes Online*. Spiegel & Grau, 2016, ISBN : 978-0812997859.
- [2] ALONSO, P. et GUITON, A., « Evgeny Morozov : “Les technologies sont des concentrés d’idéologies” », *Libération*, avr. 2015. adresse : http://www.liberation.fr/eclairs/2015/04/20/les-technologies-sont-des-concentres-d-ideologies_1254606.
- [3] ANNETT, J. et DUNCAN, K. D., « Task analysis and training design. », 1967.
- [4] BENKLER, Y., « The realism of cooperativism », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 14, page 184, ISBN : 9781682190623.
- [5] BOLLIER, D., « From open access to digital commons », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 13, page 184, ISBN : 9781682190623.
- [6] BONACCORSI, A. et ROSSI, C., « Why open source software can succeed », *Research Policy*, tome 32, numéro 7, pages 1243–1258, 2003, ISSN : 00487333. DOI : [10.1016/S0048-7333\(03\)00051-9](https://doi.org/10.1016/S0048-7333(03)00051-9).
- [7] BRAY, I. K., *An Introduction to Requirements Engineering*. Addison-Wesley, 2002, ISBN : 0201767929.
- [8] BRIA, F., « Public policies for digital sovereignty », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 37, page 184, ISBN : 9781682190623.
- [9] CHARAUDEAU, P., « Pour une interdisciplinarité “focalisée” dans les sciences humaines et sociales », *Questions de communication*, tome 17, 2010. DOI : [10.4000/questionsdecommunication.385](https://doi.org/10.4000/questionsdecommunication.385).

- [10] CHEN, L. et NUGENT, C., « Ontology-based activity recognition in intelligent pervasive environments », *International Journal of Web Information Systems*, tome 5, numéro 4, pages 410–430, 2009, ISSN : 1744-0084. DOI : [10.1108/17440080911006199](https://doi.org/10.1108/17440080911006199).
- [11] CHERRY, M. A., « Legal and governance structures build to share », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 38, page 184, ISBN : 9781682190623.
- [12] CRAWFORD, M. B., *The World Beyond Your Head : On Becoming an Individual in an Age of Distraction*. Allen Lane, 2015, ISBN : 978-0670066391.
- [13] CROOK, R., INCE, D. et NUSEIBEH, B., « Modelling access policies using roles in requirements engineering », *Information and Software Technology*, tome 45, numéro 14, pages 979–991, 2003, ISSN : 09505849. DOI : [10.1016/S0950-5849\(03\)00097-1](https://doi.org/10.1016/S0950-5849(03)00097-1).
- [14] DEPARTMENT OF DEFENSE, « Trusted computer system evaluation criteria », Department of Defense, rapport technique, 1985, pages 1–116. arXiv : [arXiv:1011.1669v3](https://arxiv.org/abs/1011.1669v3). adresse : <http://csrc.nist.gov/publications/history/dod85.pdf>.
- [15] *Dépistage précoce de la Maladie d’Alzheimer – Protocole de recherche du Projet DEI*, Institut universitaire de gériatrie de Montréal, 2016.
- [16] DUTIL, É., BOTTARI, C., VANIER, M. et GAUDREAU, C., *Profil des AVQ – Description de l’outil*. Montréal, Québec : Les Editions Emersion, 2005, page 45, ISBN : 2-923118-07-3.
- [17] —, *Profil des AVQ – Guide de l’utilisateur, mise en situation*. Montréal, Québec : Les Editions Emersion, 2005, page 73, ISBN : 2-923118-06-5.
- [18] EUZEN, P., « “Pédale ou crève” : dans la peau d’un livreur Foodora », *Le Monde*, mai 2017. adresse : https://lemonde.fr/entreprises/article/2017/05/26/pedale-ou-creve-dans-la-peau-d-un-livreur-foodora_5133978_1656994.html.
- [19] EVANS, E., *Domain-Driven Design : Tackling Complexity in the Heart of Software*. Addison-Wesley Professional, 2003, page 560, ISBN : 0-321-12521-5.
- [20] FELTUS, C., « Aligning Access Rights to Governance Needs with the Responsibility MetaModel (ReMMo) in the Frame of Enterprise Architecture », thèse de doctorat, Université de Namur, Belgique, 2014, pages 1–258, ISBN : 9782870378434. adresse : <http://dial.academielouvain.be/handle/boreal:144013>.

- [21] FERRAIOLO, D. F. et KUHN, D. R., « Role-Based Access Control », Dans *15th NIST-NCSC National Computer Security Conference*, 1992, pages 554–563, ISBN : 1580533701. DOI : [10.1109/2.485845](https://doi.org/10.1109/2.485845). arXiv : [0903.2171](https://arxiv.org/abs/0903.2171).
- [22] FERRAIOLO, D. F., KUHN, D. R. et CHANDRAMOULI, R., « Proposed NIST standard for Role-Based Access Control », *ACM Transactions on Information and System Security (TISSEC)*, tome 4, numéro 3, pages 224–274, 2001, ISSN : 00200190. DOI : [10.1145/501978.501980](https://doi.org/10.1145/501978.501980).
- [23] *Formulaire d'information et de consentement pour participants au projet DEI*, Institut universitaire de gériatrie de Montréal, 2014.
- [24] FRIARD, O. et GAMBA, M., « BORIS : a free, versatile open-source event-logging software for video/audio coding and live observations », *Methods in Ecology and Evolution*, tome 7, numéro 11, pages 1325–1330, 2016, ISSN : 2041210X. DOI : [10.1111/2041-210X.12584](https://doi.org/10.1111/2041-210X.12584).
- [25] FRIEDMAN, G., « Workers without employers : shadow corporations and the rise of the gig economy », *Review of Keynesian Economics*, tome 2, numéro 2, pages 171–188, avr. 2014. DOI : [10.4337/roke.2014.02.03](https://doi.org/10.4337/roke.2014.02.03).
- [26] FUCHS, L., PERNUL, G. et SANDHU, R., « Roles in information security - A survey and classification of the research area », *Computers and Security*, tome 30, numéro 8, pages 748–769, 2011, ISSN : 01674048. DOI : [10.1016/j.cose.2011.08.002](https://doi.org/10.1016/j.cose.2011.08.002).
- [27] GAZZALEY, A. et ROSEN, L. D., *The Distracted Mind : Ancient Brains in a High-Tech World*. The MIT Press, 2016, ISBN : 978-0262034944.
- [28] GOODMAN, M., *Future Crimes : Inside the Digital Underground and the Battle for our Connected World*. Transworld Publishers, 2015, ISBN : 0385539002.
- [29] GREIF, S., *Understanding Meteor Publications & Subscriptions*, jan. 2014. adresse : <https://www.discovermeteor.com/blog/understanding-meteor-publications-and-subscriptions/>.
- [30] HALL, J. G., RAPANOTTI, L. et JACKSON, M. A., « Problem oriented software engineering : Solving the package router control problem », *IEEE Transactions on Software Engineering*, tome 34, numéro 2, pages 226–241, 2008, ISSN : 00985589. DOI : [10.1109/TSE.2007.70769](https://doi.org/10.1109/TSE.2007.70769).

- [31] HAMONET, C. et BÉGUÉ-SIMON, A., « Utilisation du concept de la classification de WOOD dans l'enseignement du handicap et de la réadaptation aux étudiants et professionnels de la santé. Communication au colloque du C.T.E.N.R.H.I. "Classification internationale des handicaps. Du concept à l'application." », *Réadaptation*, tome 355, pages 20–22, 1988.
- [32] HARDIN, G., « The tragedy of the commons », *Science*, tome 162, numéro 3859, pages 1243–1248, 1968.
- [33] HARRIS, T., « How technology is hijacking your mind – from a magician and google design ethicist », mai 2016. adresse : <https://journal.thriveglobal.com/how-technology-hijacks-peoples-minds-from-a-magician-and-google-s-design-ethicist-56d62ef5edf3>.
- [34] HASSENZAHN, M., BEU, A. et BURMESTER, M., « Engineering joy », *IEEE Software*, tome 18, numéro 1, pages 70–76, 2001, ISSN : 07407459. DOI : 10.1109/52.903170.
- [35] HASSENZAHN, M., BURMESTER, M. et KOLLER, F., « AttrakDiff : ein fragebogen zur messung wahrgenommener hedonischer und pragmatischer qualität », Dans *Mensch & Computer 2003. Interaktion in Bewegung*, ZIEGLER, J. et SZWILLUS, G., éditeurs, 2003, pages 187–196, ISBN : 3-519-00441-0. DOI : 10.1007/978-3-322-80058-9_19.
- [36] HASSENZAHN, M., PLATZ, A., BURMESTER, M. et LEHNER, K., « Hedonic and ergonomic quality aspects determine a software's appeal », *Proceedings of the SIGCHI conference on Human Factors in Computing Systems CHI 00*, tome 2, numéro 1, pages 201–208, 2000, ISSN : 00144886. DOI : 10.1145/332040.332432.
- [37] HILL, S., « How the un-sharing economy threatens workers », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 9, page 184, ISBN : 9781682190623.
- [38] HIPPEL, E. von et KROGH, G. von, « Open source software and the “private-collective” innovation model : issues for organization science », *Organization Science*, tome 14, numéro March, pages 209–223, 2003. DOI : 10.1287/orsc.1050.0136.
- [39] IGBARIA, M., SCHIFFMAN, S. J. et WIECKOWSKI, T. J., « The respective roles of perceived usefulness and perceived fun in the acceptance of microcomputer technology », *Behaviour & Information Technology*, tome 13, numéro 6, pages 349–361, nov. 1994. DOI : 10.1080/01449299408914616.

- [40] JACKSON, M., *Problem Frames and Methods : Analysing and Structuring Software Development Problems*. Addison-Wesley, 2001, ISBN : 020159627X.
- [41] JAGODZIŃSKI, Ł. (2016). Astronomy Documentation v2.x, adresse : <http://jagi.github.io/meteor-astronomy/v2> (visité le 28/02/2017).
- [42] JOHN, B. E. et KIERAS, D. E., « The GOMS family of user interface analysis techniques : Comparison and contrast », *ACM Transactions on Computer-Human Interaction (TOCHI)*, tome 3, numéro 4, pages 320–351, 1996.
- [43] JOHNSON, H. et JOHNSON, P., « Task knowledge structures : psychological basis and integration into system design », *Acta psychologica*, tome 78, numéro 1, pages 3–26, 1991.
- [44] KIPP, M., « ANVIL A Generic Annotation Tool for Multimodal Dialogue », *Proceedings of the 7th European Conference on Speech Communication and Technology*, pages 1367–1370, 2001.
- [45] KLEIN, J. T., « Une taxinomie de l’interdisciplinarité », *Nouvelles perspectives en sciences sociales*, tome 7, numéro 1, pages 15–48, 2011. DOI : [10.7202/1007080ar](https://doi.org/10.7202/1007080ar).
- [46] KLEINER, D., « Counteranti-disintermediation », Dans *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, chapitre 12, page 184, ISBN : 9781682190623.
- [47] LALLEMAND, C., KOENIG, V., GRONIER, G. et MARTIN, R., « Création et validation d’une version française du questionnaire AttrakDiff pour l’évaluation de l’expérience utilisateur des systèmes interactifs », *Revue Européenne de Psychologie Appliquée*, tome 65, numéro 5, pages 239–252, 2015, ISSN : 11629088. DOI : [10.1016/j.erap.2015.08.002](https://doi.org/10.1016/j.erap.2015.08.002).
- [48] LESSIG, L., *CODE version 2.0*. 2006, pages 1–424, ISBN : 9780465039142. adresse : <http://codev2.cc>.
- [49] LUKIC, K., « Colonization with Love », *Share Foundation*, août 2016. adresse : <https://labs.rs/en/colonization-with-love/>.
- [50] MAYFIELD, T., ROSKOS, J. E., WELKE, S. R. et BOONE, J. M., « Integrity in automated information systems », National Computer Security Center, rapport technique September, 1991, page 130.
- [51] MAZZUCATO, M., *The Entrepreneurial State : Debunking Public vs. Private Sector Myths*. PublicAffairs, 2015, ISBN : 1610396138.

- [52] METEOR DEVELOPMENT GROUP INC. (2012). Meteor, adresse : <https://www.meteor.com/> (visité le 12/09/2016).
- [53] MORIN, E., « Sur l'interdisciplinarité », Dans *Carrefour des sciences*, Actes du Colloque du Comité National de la Recherche Scientifique Interdisciplinarité, CNRS, É. du, éditeur, 1990, pages 21–29.
- [54] MOROZOV, E., « The rise of data and the death of politics », *The Guardian*, juil. 2014. adresse : <https://www.theguardian.com/technology/2014/jul/20/rise-of-data-death-of-politics-evgeny-morozov-algorithmic-regulation>.
- [55] NOLDUS INFORMATION TECHNOLOGY. (2017). Observer XT, adresse : <http://www.noldus.com/human-behavior-research/products/the-observer-xt> (visité le 03/04/2017).
- [56] —, (2017). Viso, adresse : <http://www.noldusviso.com/> (visité le 13/05/2017).
- [57] ORAIN, G., « Sur Amazon Turk, les forçats du clic », *Le Monde*, mai 2017. adresse : http://www.lemonde.fr/pixels/article/2017/05/22/les-damnes-de-la-toile_5131443_4408996.html.
- [58] OSTROM, E., *Governing the Commons : The Evolution of Institutions for Collective Action (Political Economy of Institutions and Decisions)*. Cambridge University Press, 1990, ISBN : 0521405998.
- [59] PÉRÈS, K., HELMER, C., AMIEVA, H., ORGOGOZO, J.-M., ROUCH, I., DARTIGUES, J.-F. et BARBERGER-GATEAU, P., « Natural history of decline in Instrumental Activities of Daily Living performance over the 10 years preceding the clinical diagnosis of dementia : A prospective population-based study », *Journal of the American Geriatrics Society*, tome 56, numéro 1, pages 37–44, jan. 2008. DOI : [10.1111/j.1532-5415.2007.01499.x](https://doi.org/10.1111/j.1532-5415.2007.01499.x).
- [60] PERKEL, J. M., « Annotating the scholarly web », *Nature*, tome 528, pages 153–154, déc. 2015. DOI : [10.1038/528153a](https://doi.org/10.1038/528153a).
- [61] (2014). Platform Cooperativism Consortium, adresse : <https://platform.coop/> (visité le 01/06/2017).
- [62] RANDOLPH, J. (2016). Serrurier, a declarative extension for methods access control in Astronomy using decorators, adresse : <https://github.com/sveinburne/serrurier#head> (visité le 28/02/2017).

- [63] RESWEBER, J.-P., « Les enjeux de l'interdisciplinarité », *Questions de communication*, tome 19, pages 171–200, 2011. DOI : [10.4000/questionsdecommunication.2661](https://doi.org/10.4000/questionsdecommunication.2661).
- [64] RIFKIN, J., *The Third Industrial Revolution : How Lateral Power Is Transforming Energy, the Economy, and the World*. St. Martin's Griffin, 2013, ISBN : 0230341977.
- [65] SANDHU, R., COYNE, E., FEINSTEIN, H. et YOUMAN, C., « Role-based access control models », *Computer*, tome 29, numéro 2, pages 38–47, 1996, ISSN : 00189162. DOI : [10.1109/2.485845](https://doi.org/10.1109/2.485845).
- [66] *Ours To Hack And Own : The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. OR Books, 2016, page 184, ISBN : 9781682190623.
- [67] SCHOLZ, T., éditeur, *Digital Labor : The Internet as Playground and Factory*, Routledge, 2012, ISBN : 0415896959.
- [68] SCHOLZ, T., *Platform Cooperativism vs. the Sharing Economy*, 2014. adresse : <https://medium.com/@trebors/platform-cooperativism-vs-the-sharing-economy-2ea737f1b5ad>.
- [69] TARNOFF, B., « America has become so anti-innovation – it's economic suicide », *The Guardian*, mai 2017. adresse : <https://www.theguardian.com/technology/2017/may/11/tech-innovation-silicon-valley-juicero>.
- [70] TECHNOLOGIE ERGONOMIE APPLIQUÉES. (2017). Captiv, adresse : <http://teargo.com/site/en/products/manufacturers/tea/captiv-software> (visité le 03/04/2017).
- [71] VAN DER VEER, G. C., LENTING, B. F. et BERGEVOET, B. A., « GTA : Groupware task analysis—Modeling complexity », *Acta psychologica*, tome 91, numéro 3, pages 297–322, 1996.
- [72] VINCENT, N. A., *Moral Responsibility*. Springer Netherlands, 2011, tome 27, ISBN : 978-94-007-1877-7. DOI : [10.1007/978-94-007-1878-4](https://doi.org/10.1007/978-94-007-1878-4).